Speech-driven Animation using Multi-modal Hidden Markov Models

Gregor Otto Hofer

Doctor of Philosophy

Centre for Speech Technology Research

School of Informatics

University of Edinburgh

2009

Abstract

The main objective of this thesis was the synthesis of speech synchronised motion, in particular head motion. The hypothesis that head motion can be estimated from the speech signal was confirmed. In order to achieve satisfactory results, a motion capture data base was recorded, a definition of head motion in terms of articulation was discovered, a continuous stream mapping procedure was developed, and finally the synthesis was evaluated. Based on previous research into non-verbal behaviour basic types of head motion were invented that could function as modelling units. The stream mapping method investigated in this thesis is based on Hidden Markov Models (HMMs), which employ modelling units to map between continuous signals. The objective evaluation of the modelling parameters confirmed that head motion types could be predicted from the speech signal with an accuracy above chance, close to 70%. Furthermore, a special type of HMM called trajectory HMM was used because it enables synthesis of continuous output. However head motion is a stochastic process therefore the trajectory HMM was further extended to allow for non-deterministic output. Finally the resulting head motion synthesis was perceptually evaluated. The effects of the "uncanny valley" were also considered in the evaluation, confirming that rendering quality has an influence on our judgement of movement of virtual characters. In conclusion a general method for synthesising speech-synchronised behaviour was invented that can applied to a whole range of behaviours.

Acknowledgements

First I would like to express my gratitude to my supervisor Dr. Hiroshi Shimodaira for always finding time to meet with me, his patience with my questions, expert guidance, constructive criticism when necessary but also his encouragement to keep on working. In addition I would like to thank my second supervisor Prof. Steve Renals who encouraged me to do a PhD at CSTR in the first place.

I would like to also thank Dr. Junichi Yamagishi for our fruitful discussions and exchange of ideas that had a significant positive impact on the quality of my work.

Furthermore I would like to thank all of the members of CSTR for making this place such an interesting and relaxed place to work at. In particular I would like to thank Dr. Simon King, for his advice on everything from the coffee machine to distributed processing and Michael Berger for being such a great office mate.

I would like to also thank all my friends and family for their love and care. Finally I would like to thank my dear Fiancee Daniela for her unlimited patience and support while I was finishing my PhD.

Declaration

I declare that this thesis was composed by myself, that the work contained herein is my own except where explicitly stated otherwise in the text, and that this work has not been submitted for any other degree or professional qualification except as specified.

(*Gregor Otto Hofer*)

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

Animating computer graphics characters (cg) is a time consuming process that involves many repetitive steps. At each frame the animator has to specify the facial expression and position of the character. As most characters are not only moving but also talking, one of the most labour intensive processes for the animator is the synchronisation of the character's behaviour with speech. The animator is manually marking regions of interest in the speech signal and constructs the corresponding animations. Constructing animations by hand means setting the values of up to 70 parameters per frame. It is therefore highly desirable to automate at least part of that process.

The way a character expresses itself is highly dependent on the personality and the emotional state that the animator is trying to give to the character. Most cg characters use real speech from an actor where the persona that the actor is portraying is transferred to the character. The persona influences the movement style of the character on many levels, like the speed of the motion or, in speech animation, the specific idiosyncrasies of mouth movements by each person and character.

In addition to emotional variation, getting consistent performance capture, the non-linear mapping between speech and motion, and the high dependency on personality of the character's motion are the main aspects that make automating the process of animation difficult. Still, even automating parts of the animation process can be an enormous time saver and lets the animator concentrate on more artistic work.

Using current speech technology it is possible to automatically generate acceptable lip animation, but speech animation is more than just the movement of the lips. It encompasses a wide range of behaviour that has significant impact on the perception and intelligibility of the speech. Take head motion for example: The way a person moves their head during speech helps us understand what is being said. It is therefore important when trying to automate speech animation, that non-verbal behaviour is also included.

For an automatic speech driven animation system or even a semiautomatic one to be useful, the representation of the motion has to be in human-readable form. The animator needs to be able to change the output of such a system to fit his or her needs, as the output can not always be expected to be optimal. The form of the output is relatively clear for lip animation, as phonemes are widely used and understood. Each phoneme would produce a quite distinct lip shape but when it comes to other types of behaviour the representation of motion is much less well understood. It is therefore imperative to develop an appropriate representation or modelling unit that can easily be manipulated by a human. In addition it is desirable to be able to learn a specific movement style from an actor and transfer it to the character.

One of the reasons for the high prevalence of manual labour in the production of good speech animation is the non-linear mapping between the audio and the visual. The relationship of the way a human moves and the sounds that he makes cannot be expressed in a simple linear mapping. The further the motion is removed from the direct articulation the mapping becomes more non-linear and stochastic. The same utterance can be expressed in many different ways, and still perceived as natural, making the problem a one to many mapping. Despite this non-linearity, we perceive a clear synchrony between motion and speech, or in other words we are highly perceptive to the absence of this synchrony.

The non-linear properties of the mapping makes a rule based solution unlikely. Therefore machine learning will be applied to the problem of automatic speech-driven animation. In particular generating an animation from speech can be seen as exploiting a learned mapping from speech to motion. Therefore this thesis aims to develop a speech-driven animation system, that is trained on data using Hidden Markov Models (HMMs).

HMMs have been used extensively in speech recognition and recently speech synthe-

sis. These models can be useful for visual speech synthesis as well since the mapping from speech features to facial poses is many-to-many. Much of the complexity comes from contextual factors like co-articulation. An HMM can make optimal use of context of a whole utterance if properly trained. The speech synthesis community has developed methods to generate smooth trajectories from stochastic models such as HMMs. In particular the trajectory HMM developed by Tokuda and colleagues has been very successful in recent years. By having dynamic constraints on the parameter generation they are able to produce smooth output from a conventionally trained HMM (Tokuda et al. 2000). This model has previously been applied to lip motion generation by training a model on video frames. The modelling unit is the syllable (Tamura et al. 1998). Although previous studies have show some promising results, there has been no principal investigation into the utility of trajectory HMMs for animation. Since HMMs are mostly used in speech technology, there are modifications and extensions needed to make these system work reliably for speech animation. Clearly, producing animation is different from producing speech. Speech synthesis is the mapping from symbols to continuous data, where-as speech driven animation is the stream to stream mapping from audio to video. Additionally the parameter generation algorithm has no built in functionality to control the dynamic range of output trajectories and synchronisation issues between the generated motion trajectory and the audio signal have to be solved. Finally, trajectory HMM synthesis is deterministic which might not be desirable in motion synthesis, as many alternate motion sequences are acceptable. In speech synthesis prosody is an example where many possible alternatives exist for any given utterance, likewise many different movement patterns exist for any given utterance.

Most previous research in the area of speech driven animation has focused on generating appropriate lip motion but speech synchronised behaviour includes other forms of movement as well. In this thesis the focus will be on head motion because any natural speaking animation includes head motion. Head motion not only increases perceived naturalness but also improves the speech intelligibility (Munhall et al. 2004). Furthermore very little research has been done in the automatic generation of head motion and it remains a very challenging problem. Kendon (2004) and others have put forth the idea that head motion can be viewed as a gesture, which can be related to speech production. In addition to conveying meaning, head motion can also be viewed as part of the articulation. These concepts can be further illuminated by the hypothesis that speech and gesture originate from a single idea. An attempt to place the formu-

lation of an idea in relation to head motion can been seen in Figure 1.1. Although non-verbal and verbal behaviour work on a similar level head motion is probably part of both. Hadar et al. (1985) based on their experiments view head motion as having both semantic and motoric functions. The motoric functions are defined as low level expressions that happen in accordance with other body motion. The brain is only capable of sending a limited amount of signals in parallel. If the brain tells the articulators to move in a certain fashion, any other motion that happens while we speak will have a strong relationship to the articulation. In particular it is hypothesised that motoric head motion is part of the articulation. The head motion that will be synthesised in this thesis will be based upon the assumption that part of the head motion during speech can be predicted from the articulation.

1.2 Objectives

The central theme of this thesis is the improvement of automatic speech animation. A general framework for mapping between the speech and motion signals is presented. In particular head motion synthesis will be used to demonstrate the feasibility of the framework.
In summary the contributions of this thesis are:

- It is demonstrated that it is feasible to generate head motion from speech without any transcriptions, using just low level parameters to perform the mapping.
  - This successful mapping supports the theory that head motion is part of the articulation.
- An optimal modelling unit for head motion in terms of mapping accuracy is discovered.
- Trajectory HMMs are applied to head motion synthesis to generate smooth output.
  - To address a specific shortcoming of trajectory HMM parameter generations a concise method for generating stochastic trajectories from an HMM is developed.
- The animated head motion is evaluated in terms of different synthesis methods

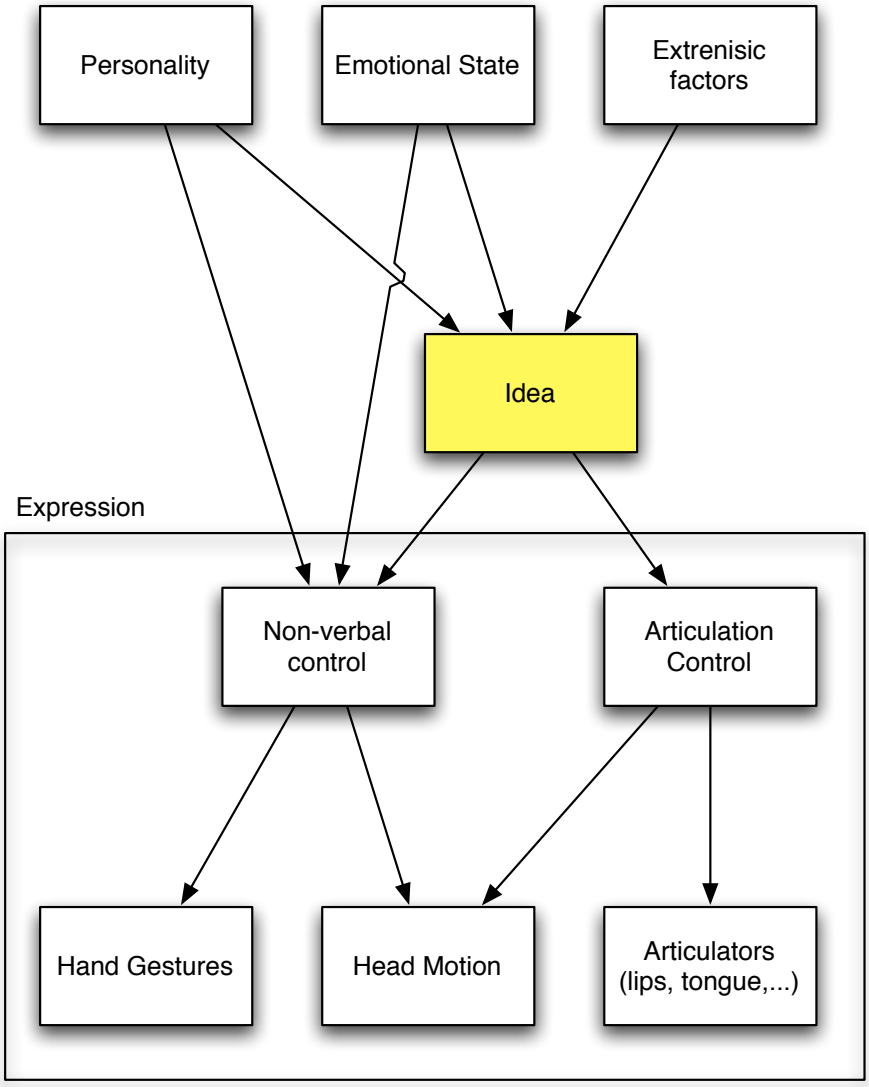

Figure 1.1: From an idea to an expression. An idea is formed in the brain given the emotional state, personality factors, and other external stimuli. To express this idea several communication channels are used but the control signal for all these channels originate from a single source. The brain can only send a limited number of signals in parallel, making all the communication channels are highly correlated. Therefore Head motion is related to both non verbal and verbal behaviour because they both originate from the same idea.

and rendering methods, which confirms that there is significant interaction between the looks of a character, the quality of the animation, and the perceived naturalness of the whole scene.

Nevertheless it is not straightforward to take an existing method and apply it to a new problem, therefore this thesis aims to extend the trajectory HMM framework to animation. The utility of the trajectory HMM in mapping from one domain to another, namely speech and motion and generating animation from this mapping will be explored. This thesis will not investigate the mapping from other information, like emotions, as its assumed that speech carries enough information to produce synchronised behaviour. Also the semantics of movements will not be investigated either. Only the head motion that is assumed to be part of the articulation will be considered. Ultimately, the presented framework can be applied to any form of animation, but this thesis will limit its scope to lip motion and head motion. In addition the head motion investigated is limited to motoric motion, head motion that conveys meaning will not be investigated.

From speech synthesis it is known that HMM-based synthesis is highly dependent on its modelling units. When modelling movement that is further removed from the speech production process like head motion, the choice of unit is not straightforward. This thesis will try to develop a practicable, human readable, and reliable modelling unit for head motion. Furthermore a concise method for generating stochastic trajectories from an HMM will be proposed. Finally the evaluation of head motion with little dependency to the speech signal will be investigated.

1.3 Thesis Structure

Chapter 2 gives a review of speech animation and a detailed description of head motion in the context of speech.

Chapter 3 describes the data collection process and also gives statistics on the data.

Chapter 4 describes the trajectory HMM framework used in the research.

Chapter 5 formally defines the problem of speech-driven animation as a mapping problem and theoretically explains the solution that has been developed as part of this thesis.

Chapter 6 describes how the mapping procedure can be applied to motion synthesis. Both lip motion and head motion generation are explained. It also gives detail on the developed modelling unit.

Chapter 7 describes the method and results of the perceptual evaluation.

Chapter 8 summaries the main achievements of the thesis and gives future directions.

1.4 Publications

Hofer. G., Yamagishi, J., and Shimodaira, H. *Speech-driven lip motion generation with a trajectory HMM*. In Proc. Interspeech 2008, Brisbane, Australia, September 2008.

Hofer, G., and Shimodaira, H. Automatic Head Motion Prediction from Speech Data. In Proc. Interspeech, Antwerp, Belgium, 2007.

Hofer, G., Shimodaira, H., and Yamagishi, J. Speech-driven Head Motion Synthesis based on a TRajectoy Model. Poster at Siggraph, Sand Diego, 2007.

Hofer, G., Shimodaira, H., and Yamagishi, J. Lip Motion synthesis using a context dependent trajectory Hidden Markov Model. Poster at SCA, Sand Diego, 2007.

Hiroshi Shimodaira, Keisuke Uematsu, Shin'ichi Kawamoto, Gregor Hofer, and Mitsuru Nakai. Analysis and Synthesis of Head Motion for Lifelike Conversational Agents. MLMI2005, Edinburgh.

Chapter 2

Speech-animation Review

2.1 Introduction

When most people hear the term speech-animation they think about lip synchronisation. Although it is not the main aspect of this thesis, a short review of common methods follows. The main aspect of this thesis is the animation of head motion. Head motion will be defined and its animation will be motivated in this chapter. Moreover, previous attempts to animate head motion automatically will be reviewed.

2.2 Lip Synchronisation

Previous approaches of lip synchronisation can be characterised by the input to the system. Many systems use text and the corresponding phoneme string as input and then use concatenation (Graf et al. 2002), dominance functions (Cohen & Massaro 1993, 1990) or trajectory generation (Tamura et al. 1998) to produce the desired animation. Other approaches use parameterised speech directly as input and then use formant analysis (Hong et al. 2002, Wen et al. 2001), linear regression (Hsieh & Chen 2006, 2005), or probabilistic modelling (Brand 1999, Nakamura 2002) to generate the appropriate motion. The choice between speech or text input depends largely on the application. A dialogue system where the spoken text is known, will most likely use a text based approach. An application that has to deal with unknown speech, in particular applications that have to run fast and in real time, like rapid prototyping

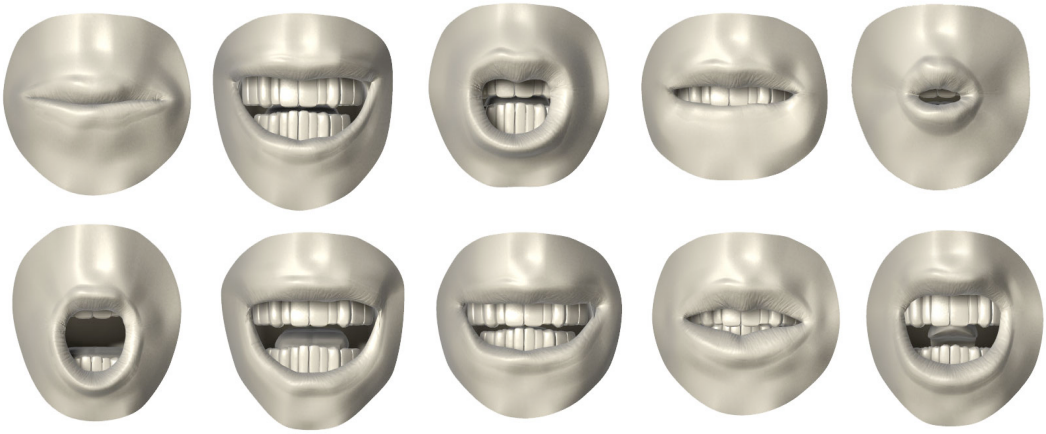

Figure 2.1: The Preston-Blair viseme set, specified for Disney as shown by (Martin 2006).

for games and movies, will opt for speech based input. However, speech is usually described in terms of phonemes, which are sub-syllable units. Many previous approaches have used visemes (visual counterpart of a phoneme) or mouth shapes to create a mapping between the speech and the animation (Cao et al. 2005). Because of co-articulation, where the previous and next mouth shape influence how the current mouth shape should look like, it is not enough to animate using a sequence of isolated mouth shape. To deal with the problem of co-articulation many previous approaches have utilised rules(Cohen et al. 2002) or statistical techniques (Ezzat et al. 2002, Chang & Ezzat 2005). In addition speech rate, defined as the number of phonemes during a given time interval, and loudness changes need to be addressed for successful lip synchronisation.

One interesting aspect of visual speech synthesis is the choice of modelling unit. Most of the previous approaches employed visemes. There hare are a number of viseme sets defined but as an example the set defined by Disney is quite well known. It consists of 9 visemes plus rest position (Martin 2006). The approach outlined in this thesis relies quite heavily on the modelling unit and therefore the choice of unit for lip synchronisation will also be investigated.

For an in depth review of audio-visual speech synthesis please see Bailly et al. (2003).

2.3 Perception of head motion

The main application of the techniques developed in this thesis is head motion synthesis and therefore we will go into more detail in its review. Head motion is a prominent aspect of our communication effort. We are constantly moving our head, either while talking or while listening. It sends signals to other people but it can also send signals to oneself. Generally motion, but in particular head motion can be seen as smoothing the interaction with other people. Munhall et al. (2004) found that head movement increased the intelligibility of speech. It is therefore important to send the right signal when animating a character's head motion as we as humans are very aware of it. There is evidence that people have specialised processing for the face called the fusiform face area (FFA) (Kanwisher et al. 1997). This is evidence that people pay special attention, and have a lot of exposure to the area around the face. It is therefore important to animate the head in a plausible way. Since people have a lot of experience of head motion, the effect of the uncanny valley (Mori 1970) could be increased as well if the animation is unnatural. However, before head motion can be animated, it first has to be defined and then placed in context of other communication channels.

2.4 Head Motion as Gesture

There is very little research done specifically on head motion therefore it is proposed to view head motion in the context of gesture research. Gestures are defined as movement accompanying speech, McNeill (2005) describes them as fuelling thought and speech, integrated on the cognitive, and biological level. The result of a single mental process, some gestures display synchrony with speech. Different gesticulations have different amount of synchrony with speech. Therefore researchers have attempted to group types of gestures into categories that differentiate by the amount of synchrony or presence of speech. Kendon (2003, 2004) classified gestures into various categories:

- **Gesticulation** is directly related to the speech. A hand motion indicating upward would be part of this category but also beat gestures are included.
- **Speech-linked gestures** are part of an utterance, occupying a grammatical slot.
- **Emblems** are common signs, such as thumbs-up for "OK".

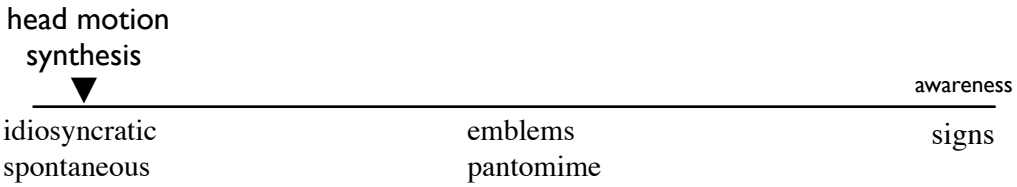

Figure 2.2: The proposed location of head motion on Kendon's continuum along the axis of awareness.

- In **pantomime** a gesture coneys a story.
- in **sign language** gestures are lexical words. The language has its own linguistic structure.

Although Kendon was primarily concerned with hand motion, it is useful to think of head motion along these categories. Since the differentiation takes place on the level of awareness of the gesture it is straightforward to translate adapt these categories for head motion. For example, we are probably fully aware of a head nod that should convey agreement but might be less aware of a nod that is part of a beat, conveying the rhythm of speech. Additionally the level of awareness of gestures and the presence of speech during gestures seem closely related. The more aware we are of the gestures, the less we produce speech. For example during 'gesticulation' speech is necessary but 'emblems' can work almost without speech, and 'signs' in sign language are produced completely without speech.

The above categories were arranged by McNeill along a continuum, termed Kendon's continuum as seen in Figure 2.2. Given this continuum the research in this thesis is primarily concerned with 'gesticulation'. It is clear that head motion exists almost on all levels but the head motion that has the closest relationship with speech is probably on the extreme left end.

Head motion can also be thought of as beats, which is the least elaborate gesture. *Beats* signal time, by moving in accordance with the rhythm of speech. Usually they are hand movements, or mere flicks of the hand but the head can also perform beat gestures. Cassell et al. (1994, 2001) found that *beats* were used to introduce new material into the discourse. A quick hand motion would coincide with the first mention of a word, with the thrust corresponding to the prosodic peak. Generally *beats* are used to place emphasis on on certain words, with McNeill using the term temporal highlighting.

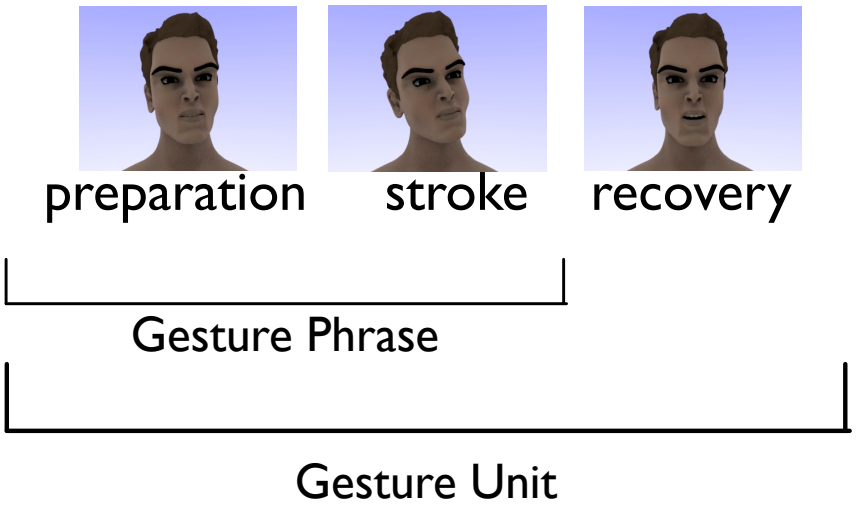

Figure 2.3: Kendon's gesture unit.

In addition to different types, the gestures have also a structure of their own. Kendon (2003, 2004) describes the phases of gestural action, which can be seen in Figure 2.3. He refers to the entire *excursion* from the moment the acting body parts (*articulators*) begin to depart from a position of relaxation to the moment when they return to one as a *gesture unit*. During such a unit, there could be multiple points when the articulators reach a position that is the furthest removed from relaxation. These points are termed *apex* and the phase of movement leading up to them is called the *stroke*. The phase of movement leading up to the stroke, is called *preparation*. Additionally the stroke might be followed by a phase where the articulator is sustained, called *hold*. The final phase when the articulator returns to rest if called *recovery*. The *gesture phrase* contains only one stroke, but also any preparation and hold. A gesture unit can consist of multiple phrases.

Following the same lines of reasoning Kendon proposes that speech is also organised into a series of packages, termed tone units. They differentiate from each other by pitch level, loudness, and pacing and they correspond to units of meaning. Gesture phrases are units of action, that are semantically coherent with the meaning expressed in the tone units. However gesture phrases do not need to be synchronised with the speech on a temporal level, meaning their start and end times do not necessarily correspond.

It is an interesting proposition to divide gestures into sections, although it is not entirely clear, where the boundaries of a gesture could be when only looking at the movement.

Kendon mostly describes units that are semantically relevant but he does not propose any units that do not convey a specific concept or meaning. Speakers produce many gestures that bear not semantic relationship to what is being said but such movement might still be important, as it increases the engagement with the speaker, like beat gestures.

2.5 Head motion during articulation

McNeill describes the unbreakable link between gesture and speech, and talks about psycholinguistical units that encompass both gestures and speech in his book Gesture and Thought (McNeill 2005). This link can be viewed from the communication point of view, where head motion is a gesture, but it can also be viewed in the sense that head motion plays a role in the speech production process. Generally it has been proposed that body movement is tied directly to linguistic structure. Birdwhistell (1970) proposed units of movements where lower level units combined to form higher level units. The units were kines that formed classes of kinemes. A Kineme was a group of movement that held distinct meaning. For example a head nod is a distinct kinesic unit with a range of 10 degrees, at a maximum of 0.8 to 3 degrees per 1/24 second. Head movements outside of this range and velocity were considered distinct. Although this system was not adopted by researchers, it is an early attempt to group head motion and other body movements.

Head Motion is in many ways difficult to describe in terms of speech. If you discount body posture shifts it has only three degrees of freedom, the rotational angles of the neck. Therefore the patterns exhibited during head motion are not as complex as other types of motion like hand gestures. Still, the timing of head motion carries information and acts as a primer to other modalities. Head Motion during speech has not been studied to a great extent but it is clear that a strong relationship between the two exists as has been demonstrated in a study by Rimé & Schiaratura (1991). They found that the imagery in the language decreased when subjects attempted to speak with their head fixed. There have also been some psycho-linguistic studies on the links between spoken language and head motion.

Hadar et al. (1983) conducted a number of experiments during the early 1980s that provide some insight into the patterns of Head Motion during speech. In their stud-

ies head motion was recorded with a polarised-light goniometer while a subject was speaking to an interviewer. They separated the recorded motion is five types: rapid (RM), ordinary(OM), slow(SM), still(Z), and postural shift (PS). A movement was defined as ”distinct” if it was rapid, ordinary, or a postural shift. Over a speaking period of 6.3 minutes they recorded 199 distinct movements. These movements were analysed in terms of stress and juncture on the sentence level. Distinct movements correlated well with peaks in the loudness but the scale of the peaks did not correlate with movement type. Further they found that pauses in the speech signal were usually accompanied stillness and slow movements. In another study Hadar et al. (1984) investigated head motion during speech dysfluencies. Rapid movements were found to occur many times after short pauses. The authors hypothesise that head motion can only be viewed in terms of grammatical properties but may also prime the articulators for action by generating a basic level of muscular response. A similar motoric explanation was given in another study by Hadar et al. (1985) where postural shifts were found to occur frequently between sentences or clauses, providing an “external reference”, during silence for the co-ordination of the movements of the articulators.

It is obvious that we move our head a lot during speech but there are few studies that have investigated this movement in detail. McClave (2000) investigated the linguistic functions of head motion. Two friends were filmed for an hour conversing about a topic of their choice. Movements were matched to each spoken syllable or silence. It was found that lateral movements of the speaker’s head correlate with verbalisations, expressing inclusivity, intensification, and uncertainty. In narration head orientation was found to locate a referent in abstract space. Some speaker head-nods were found to trigger back-channels.

In addition to conveying meaning, head motion could also be viewed as part of the articulation. These concepts can be further illuminated by the hypothesis that speech and gesture originate from a single source, called growth point or idea unit. Head motion can be attributed to a gesture phrase or gesture unit, which is part of the speech process. Head motion can therefore be viewed as having both linguistic functions and motoric functions. The motoric functions are a direct expression of the articulation of speech and it should be possible to predict these motoric patterns from the speech signal and use them for animation. Following Hadar et al. we hypothesise how head motion is generated by humans, having both linguistic semantic factors but also articulatory fac-

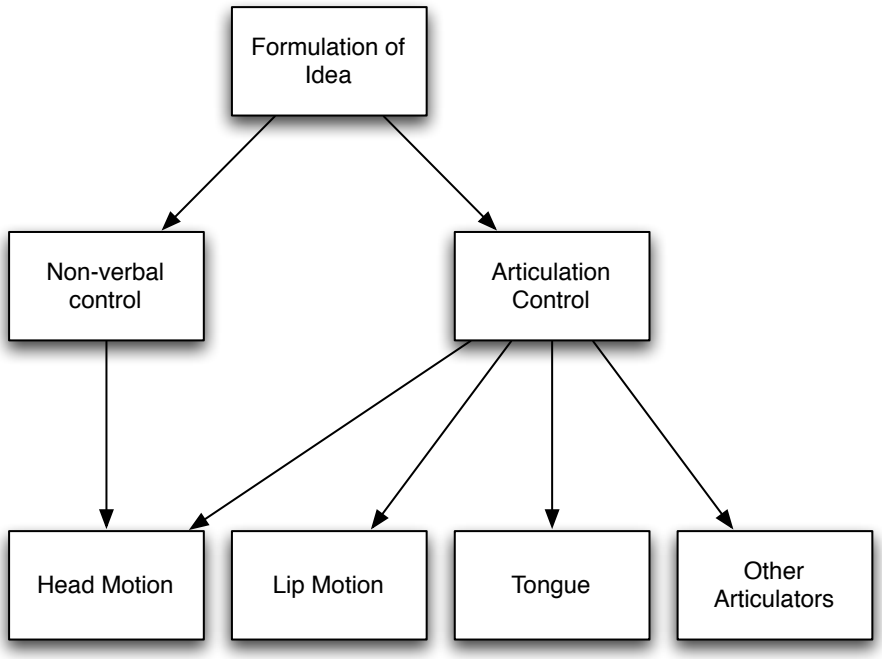

Figure 2.4: Hypothetical generation of head motion: This figure illuminates the point that head motion is influences by two more than one factor. Starting from the formulation of an idea, the motor control steers articulation and non-verbal behaviour simultaneously. When we consciously articulate the non-verbal control is influenced by the movement of the articulators. Therefore head motion can be seen has having cognitive/semantic components and articulatory components.

tors contribute to the actual movement. Figure 2.4 shows the envisioned process as hierarchical. First, it start out as an idea that is formulated into non-verbal and speech expression. Both the non-verbal behaviour and the articulation of the speech influence the head motion.

2.6 Head Motion in animation

The previous sections alluded to the fact that head motion during speech has many functions. Some examples of what is associated with head motion are: control and organize an interaction, emotional expression, lexical repairs, mark listings of alternatives, signaling interest, express inclusivity, initiating speech, accompany the rhythmic aspects of speech, and so forth. In total no less than 27 distinct functions of head mo-

tion have been identified in a literature survey by Heylen (2006). The following papers attempted to incorporate some of the identified functions into an automatic system.

One of the earliest attempts to generate head motion from speech by Cassell et al. (Cassell et al. 1994) and later by Pelachaud et al. (Pelachaud et al. 1996) was rule based. The rules were based on Hadar et al's findings. For each utterance rules are applied according to the type of utterance, specified phonemic items, stress, etc. The head motion generation was only a part in a system that attempted to synthesise a whole range of behaviour based on text mark up. They separated facial movement into phonemic, intonational, informational, and affectual determinants. Head motion was mainly used as a intonational and regulating factor in the interaction. It is interesting to see that by applying psychological findings, results can be achieved but the developed rules seem very complicated. Furthermore the interaction among them could lead to emergent unforeseen emergent behaviour. Finally as with most rule-based systems it will be hard to extend the approach or add other elements to it as it is not clear how new rules would interact with the current ones. Along similar lines DeCarlo et al. (2002) have created a rule based head motion synthesis module for their talking head RUTH.

Busso et al. (Busso et al. 2006) employed an HMM based approach in their head motion synthesis. Head poses, represented as Euler angels were clustered by the *Linde-Buzo-Gray-vector quantisation* (LBG-VQ) technique, which computes K Voronoi cells. For each cell an HMM is trained on prosodic features that are aligned with the poses in the training data. The head motion pose sequence is determined given the a prosodic feature sequence. A sequence of head motion is generated from the means of the determined clusters. Coloured noise is added to the sequence. The noise is coloured with the covariance matrix of the clusters. Spherical cubic interpolation is applied and the 3D Euler angles are interpolated in the unit sphere by using quaternion representation. The resulting signal is down sampled to 6 points per second. These key points are transformed into quaternion representation and interpolated by spherical linear interpolation, resulting in a synthesised head motion sequence.

The first criticism of Busso's work is that he clustered the head motion frame wise but then used these clusters to generate a sequence. The actual motion generation step seems very tedious and it is unclear how much of the original movement quality is retained by adding noise, interpolating, and then selecting key points from the signal, which are then interpolated one last time. Furthermore long range dependencies be-

tween the speech and motion are not taken into account as the system operates on a frame wise basis.

Sargin et al. (2007) developed a system that generates head motion from prosodic features. First HMM based clustering is performed on head motion represented as Euler angels and prosodic features separately. The correlations between the prosody and head clusters are analysed using multi stream HMMs to determine an audio-visual mapping model. The mapping model is used to generate head motion trajectories from input speech. First the pattern sequence is determined from the prosodic features. The associated Euler angles with the pattern sequence then smoothed by a filter and used to drive a talking head.

Although Sargin et al are using longer range models, it is not clear if the patterns that result from the clustering yield meaningful head motion units as it is unsupervised. Transitions between segments have to be explicitly smoothed by averaging overlapping frames. Although this might make the transition smoother, important details might be smoothed over. When looking at head motion it is clear that the transitions between movements are very important. Finally the output of the system needs to be smoothed by a filter to remove discontinuities introduced by the sampling process. Although the first derivative of the Euler angles was used in the clustering, it is not used in the motion generation.

Zhang et al. (2007) does not generate head motion directly from speech but from words that have a specific prosody associated with them. The authors identified 94 Chinese prosodic words, and studied the patterns of head nods within them. For each word peak points in the head motion are identified. The head motion is synthesised using a sine function that is modulated by the placement of the peak points. This method does not use speech but text as input, and the text needs to be manually annotated to identify the prosody words. The resulting head motion will not look very natural as head motion exhibits more complex patterns than a modulated sine wave. Furthermore, it does not use any speech features, and therefore does not take the prosody of the speaker into account. Most previous research attributed some significance to prosody and therefore excluding it from the synthesis process might have a negative effect.

Finally all of the above approaches have the problem that they use internal representations of the head motion that cannot be understood easily by humans. This makes the resulting synthesis very difficult to control, although control is desired by most

animators.

2.7 Summary

Previous literature described head motion as having two functions: First it is a gesture but second and more important head motion is part of the articulation. In another way the distinction can be made that if head motion is seen as a gesture, the link between speech and movement is on a more semantic level. If on the other hand head motion is viewed as being part of the articulation, the link is on a deeper, more fundamental level, that the same underlying brain process controls both movement and speech articulation, that produces and modulates the sound, like the tongue.

Generally if we accept that head motion is linked on these two levels, then this can be exploited by using the speech signal to determine the kinds of possible head motions. By for example employing McNeill's theories it would be possible to determine possible head motion on the semantic level, where head motion introduces new content or forms part of the descriptive gesturing. But exploiting the synchronisation on the fundamental level of articulation, should also produce possible head motion, that is not necessarily meaningful, but still synchronised with the speech. In the case of an artificial talking head, this would mean that this kind of head motion is just smoothing the interaction, making the appearance of the character more natural. Therefore in this thesis the focus is on the kind of head motion that is linked on a biological level, that is expected from a person, but is not necessarily meaningful.

Finally, if the current animation systems want to overcome the uncanny valley and present truly natural speech animation, head motion needs to be synthesised from speech and synchronised with speech. It is not just important to improve intelligibility of speech (Munhall et al. 2004, Graf et al. 2002) but to elicit a more positive response from the user. Figure 2.5 gives a representation of how a typical animation system could make use of head motion synthesis and what other processes are linked to it. What is interesting to see is that head motion could be generated given lots of different information from different parts of the system. Most current system use some sort of affectual representation to influence the motion generation, be it lip synchronisation or gesture synthesis. Head motion synthesis could make use of this information as well, but it is also possible to synthesise head motion based on just the speech given

that it also has motoric functions. The system that will be investigated in this thesis, will mostly be concerned with information that is speech related and not necessarily semantic.

Figure 2.5: Typical animation system and where head motion generation could be placed in such a system. The dashed lines represent the information that I used in this thesis to generate head motion. It fits well with Figure 1.1 where the articulation is analogous to the speech features and lip motion in the animation system.

Chapter 3

Data Collection

3.1 Introduction

The main methods employed in this thesis are data-driven and based on machine learning. Therefore training and testing data is required. Unfortunately, there is very little speech synchronous motion data in the public domain, therefore a data collection was carried out. It was decided to use a marker based optical motion capture system to track the movement of subjects while they are speaking.

3.2 Definition of Head Motion

Head motion in this thesis is defined as the rotation of the head around its axes. Of course, we are aware of the fact that body posture and the structure of the neck influence head motion in more than 3 ways, but it is not considered here. The technical limitation of the motion capture system and the animation system prevented us to simulate head motion in its full detail. Therefore the head has three degrees of freedom as shown in Figure 3.1. The rotation angles are called Euler angles and are defined as follows:

- Positive yaw is defined as a left head rotation.
- Positive pitch is an upward head nod.
- Positive roll is a left head tilt.

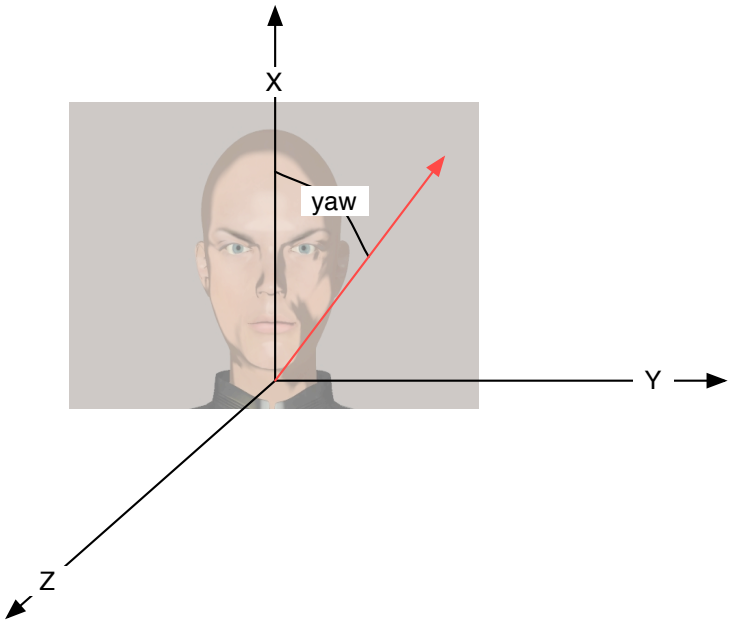

Figure 3.1: The three degrees of freedom of the head, given by the Euler angles.

3.3 Motion Capture

Optical motion capture uses image sensor data to triangulate the position of an object. Two or more cameras are calibrated to provide overlapping projections. The Qualisys ProRelfex MCU system used in this thesis is based on tracking passive markers. The markers are coated with a reflective material and attached to the subjects head and face. The systems emit infrared light which is reflected by the markers. The centroid of each marker is estimated from the infrared 2D picture that each camera takes. One camera is pictured in Figure 3.2. The actual process of motion capturing yields 3D trajectories for each Marker. The motion capture trajectories were recorded at 500 Hz. Figure 3.3 shows the layout of the capturing process.

3.3.1 Calibration and Recording

The first step in the motion capturing session is to calibrate the system by using two objects: a stationary L-shaped reference structure with four markers attached to it. The stationary L-structure defines the origin and orientation of the global co ordinate system that is to be used with the camera system. The other calibration object is called

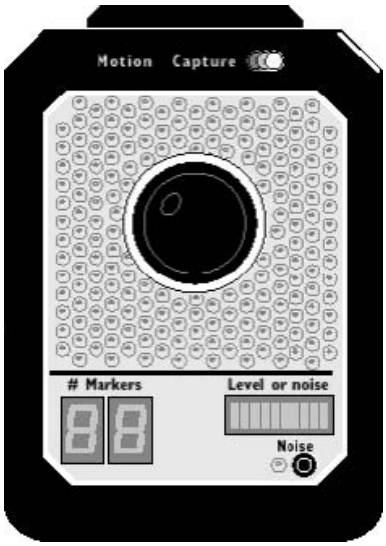

Figure 3.2: Schematic of the motion capture camera from (AB 2006). The number of markers it currently sees is indicated as well as the level of noise. Infrared LEDs are placed around the lens.

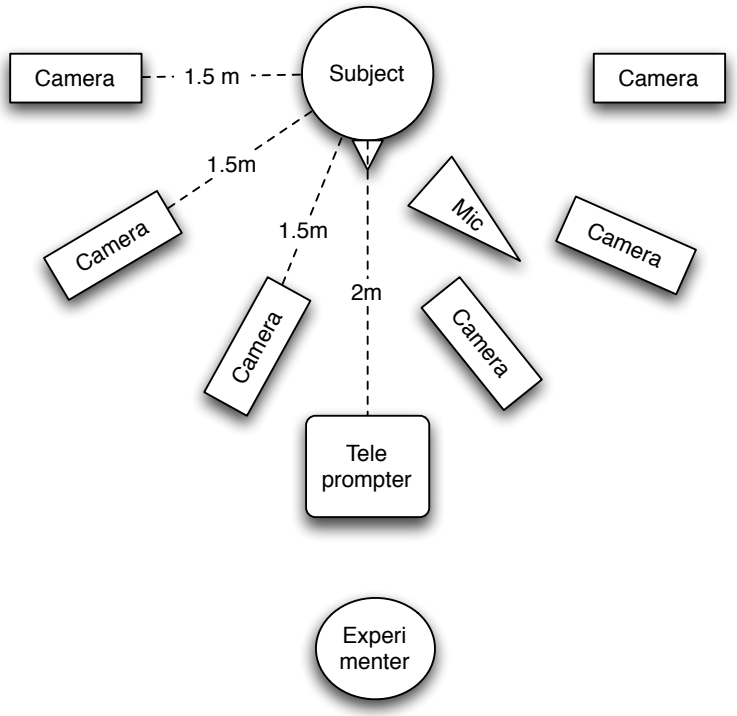

Figure 3.3: The layout of the data collection. The cameras were approximately 1.5 metres from the subject, placed in a half circle. The tele prompter was about 2 metres directly in front of the subject. The microphone was placed to the side and bottom of the subject.

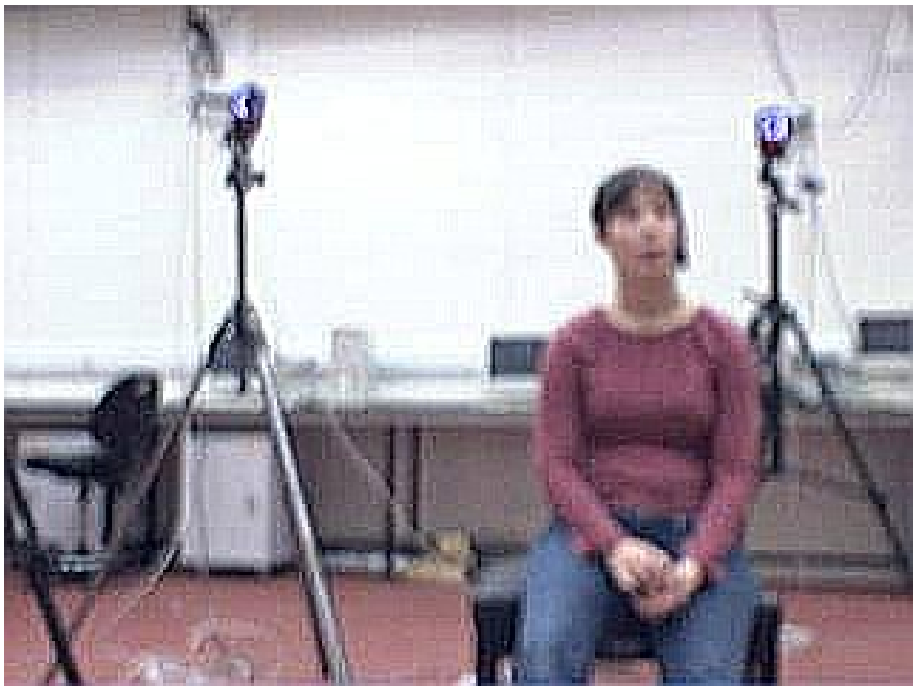

Figure 3.4: Frame from the video recording taken during data acquisition. The subject is seated among the 6 motion capture cameras.

calibration wand. It consists of two markers located a fixed distance from each other. The wand is moved in the measurement volume to generate data to determine the locations and orientations of the cameras.

Each camera has groups of infrared light emitting diodes mounted around the lens, which flash the recording frequency of 500 Hertz. The I-R light hit retro-reflective markers which return energy to the camera lens where it creates a circular reproduction on the camera's image sensor. The centre point and size of each marker is calculated in real time by a proprietary sub-pixel algorithm, providing the 2-D marker position. Figure 3.4 shows a video frame from the recordings.

3.3.2 Marker positions

In order to estimate the rotation of the head reliably, stationary points are tracked. 11 markers in total were attached to the subjects face and body. A marker was attached to each ear, the nose, and the subjects wore a rigid headband with 3 markers attached. The markers used for tracking the head motion can be seen in Figure 3.6. Additionally

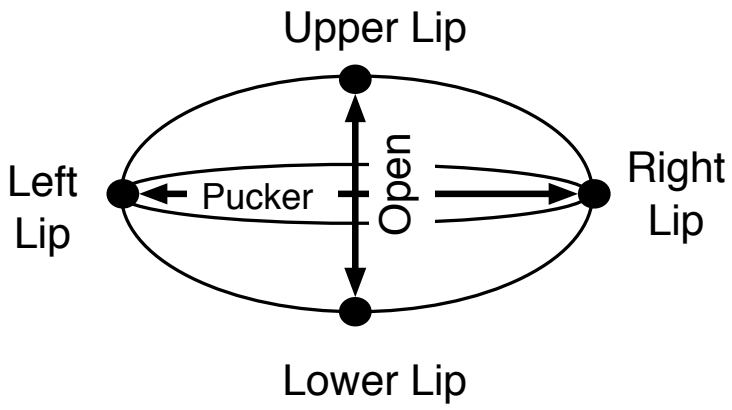

Figure 3.5: Four Markers were placed around the Lips.

4 points around the lips were tracked. The position of those can be seen in Figure 3.5.

3.3.3 Rotational Tracking of head motion

The head motion is calculated using a rigid body tracking function. First a rigid body is defined using at least four stationary points. Given the rigid body, the positional vector $\boldsymbol{P}_{origin}$ of the local coordinate system and the rotation matrix $\boldsymbol{R}$ which describes the body's rotation can be defined. Figure 3.7 shows the local coordinate system defined by the rigid body in relation to the global coordinate system. Using $\boldsymbol{R}$ a position $\boldsymbol{P}_{local}$ (e.g. x'_1, y'_1, z'_1) in the local coordinate system can be transformed to a position in the global coordinate system $\boldsymbol{P}_{global}$ (e.g. x_1, x_2, x_3). The following equation is used to transform a position:

$$\boldsymbol{P}_{global} = \boldsymbol{R} \cdot \boldsymbol{P}_{local} + \boldsymbol{P}_{origin} \tag{3.1}$$

Head motion is described using the three degrees of freedom (Euler angles), shown in Figure 3.8. The actual rotation of the body is calculated form $\boldsymbol{R}$ by expressing it in three rotation angles: roll(Θ), pitch(Φ), and yaw(Ψ). R is first described as three individual rotation matrixes: R_x, R_y, and R_z, which are individually computed by expressing the rotation in coordinates and angles. For example the rotation around the Z-axis is described in terms of x, y, and Ψ as shown in Figure 3.9.

Specifically the resulting three rotation matrixes are:

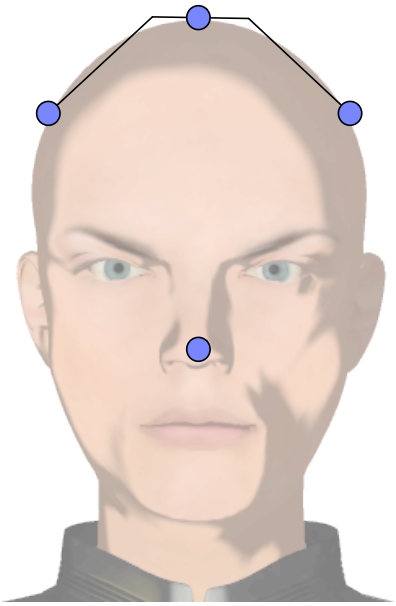

Figure 3.6: The configuration of markers attached to the head. The four markers are shown in blue.

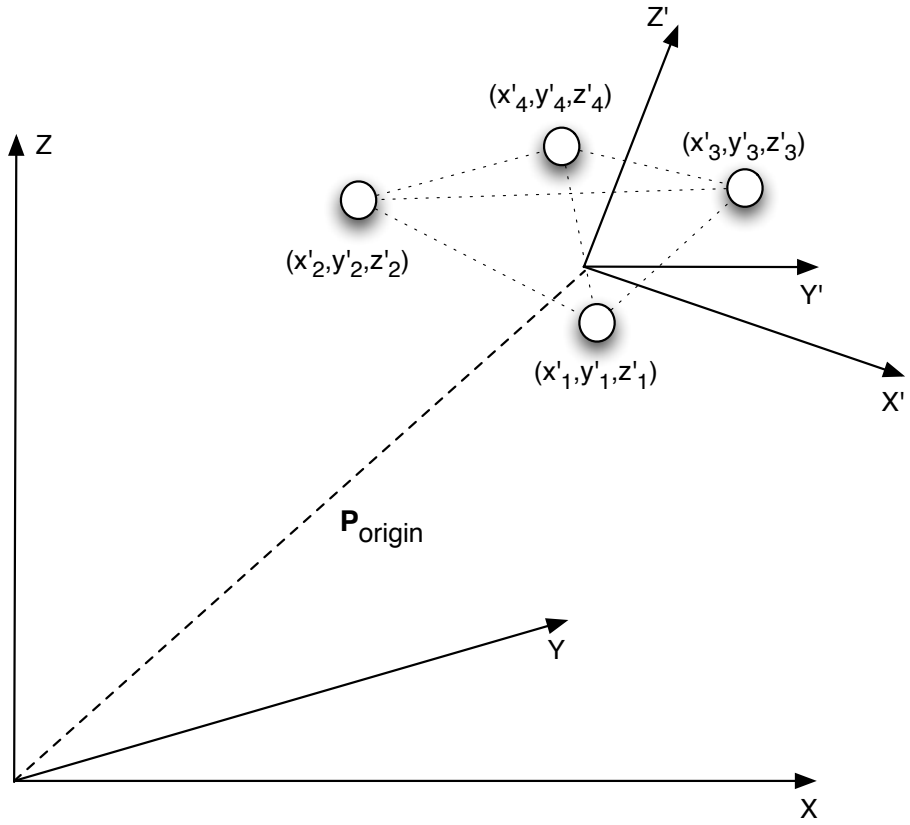

Figure 3.7: The local co ordinate system in relation to the global system.

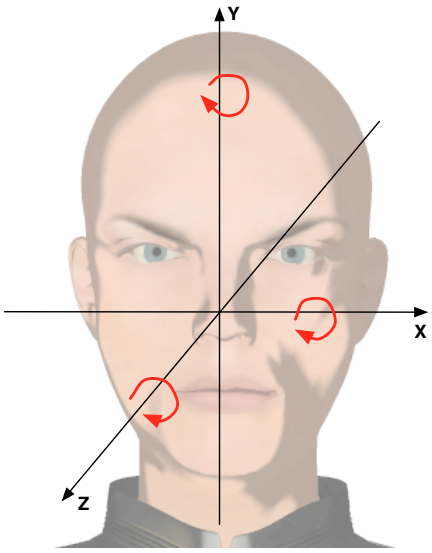

Figure 3.8: The three degrees of freedom of the head, given by the Euler angles.

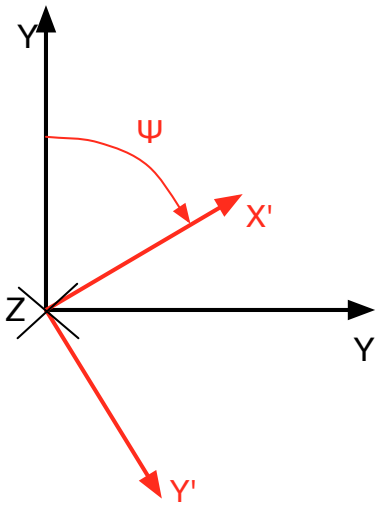

Figure 3.9: The rotation around the Z-axis as shown in the Qualisys manual (AB 2006).

$$\boldsymbol{R}_x = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\Theta & -\sin\Theta \\ 0 & \sin\Theta & \cos\Theta \end{bmatrix} \tag{3.2}$$

$$\boldsymbol{R}_y = \begin{bmatrix} \cos\Phi & 0 & \sin\Phi \\ 0 & 1 & 0 \\ -\sin\Phi & 0 & \cos\Phi \end{bmatrix} \tag{3.3}$$

$$\boldsymbol{R}_z = \begin{bmatrix} \cos\Psi & -\sin\Psi & 0 \\ \sin\Psi & \cos\Psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3.4}$$

The rotation matrix R is calculated by dot product of the individual rotation matrixes. The order of multiplication determines the application of rotation dimensions. In this case roll is applied first, then pitch, and finally yaw.

$$\boldsymbol{R} = \boldsymbol{R}_x \cdot \boldsymbol{R}_y \cdot \boldsymbol{R}_z = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{3.5}$$

The individual rotation angles are calculated from $\boldsymbol{R}$. e.g.

$$\text{Pitch:}\Phi = \arcsin(r_{13}) \tag{3.6}$$

$$\text{Roll:}\Theta = \arccos\left(\frac{r_{33}}{\cos\Phi}\right) \tag{3.7}$$

$$\text{Yaw:}\Psi = \arccos\left(\frac{r_{r11}}{\cos\Phi}\right) \tag{3.8}$$

The head motion is then described as a set of Euler angle trajectories, with each angle giving a rotation around one axis.

3.4 Database

There are very few speech synchronised motion capture databases available.

Type of data	Number of speakers	Length of data per speaker
Read speech	2	80 minutes
Free speech	12	20 minutes

Table 3.1: The amount of free and read speech data recorded for each speaker.

Selecting the appropriate material to elicit interesting head motion from speakers is not straightforward. First there is a difference between read material and free speech. Read speech is usually divided into a shorter utterances, that are maybe 10 seconds long. The subjects reads from a tele prompter or similar machine, that lets him look straight into the camera. Free speech is not read and can either be material that is freely recited like a previous heard story, or completely novel material, like the answer to the question:"What did you do last night?".

The recording situation with 6 cameras placed around the subject could also lead to less natural motion, as people become uncomfortable when they are recorded. Therefore, auditions in the actual recording setting were carried out to find optimal candidates. In total 12 speakers were recorded. Each speaker had acting training and seemed to be comfortable around recording equipment. They were recorded each for 20 minutes telling fairly tales, which they were provided with before. 10 sessions of 2 minutes were recorded. The stories were Three Little Pigs and Little Red Riding Hood. Also a short reading test was done with each speaker, checking for their word error rate and fluency. All speakers were told to behave naturally while moving their head freely and exhibiting some emotion.

The two selected speakers were the best readers, and also exhibited very varied head motion. They were judged subjectively by the author. The reason to record additional read data was to have transcribed utterances for the lip synchronisation. They were recorded reading from a TelePrompTer which was placed 2 meter in front of them. The script was the first 1000 utterances from the Arctic database provided by CMU, which consisted of sentences extracted from literary text, with good diphone coverage. The literary texts consist of 1,000 sentences selected from out-of-copyright texts from the Gutenberg project. The total amount of data is given in Table 3.1.

The data used in thesis came from Speaker 1, who was recorded for 20 minutes speaking freely and for 80 minutes reading out loud. The 1000 read utterances were divided

into 950 training utterances and 50 testing utterances. The experiments described in Section 6.2 were conducted using models trained on the 950 utterances from Speaker 1. The reported results are based on the unseen test set. The free speech data was divided into seven segments at about 3 minutes length each. Each segment consisted of a single coherent story. An exert of a story: *"This is actually a story about my Grandfather, who fought in the war. Before that he was a mountaineer, um, he climbed various mountains, he was in one of the teams that climbed K2 one of first ones. And back in those times, it was sort of colonial India, so everyone had their young man and they were caught in a landslide. So there was a terrible landslide in the night."* For the experiments presented in Section 6.6 the segments of free speech, where each segment was in effect an utterance was used to train the models. The objective recognition experiments were conducted using seven-fold cross validation with all possible permutations of training and testing segments.

3.5 Features

3.5.1 Speech Features

The audio was recorded synchronously with the motion capture data. It was not frame synchronous but the start time for both streams were the same. The speakers were recorded using an audio-technica AT 815b directional microphone at 48khz. Mel-frequencey cepstral coefficients (MFCC) are extracted from the speech waveform. They are low-dimensional frequency-domain features. A total of 12 MFCC coefficients and energy were extracted from the audio. Higher-order MFCC components provide less information than the first 12 MFCCs (Huang et al. 2001) Additionally F0 was calculated using the f0 extraction method implemented in SPTK. The frame shift for the audio features was 5 milliseconds.

3.5.2 Motion Features

The head motion features were calculated using the rotational tracking method outlined above. The lip motion features were mapped from the four tracked lip points to morph parameters. The distance between the left and right marker was mapped to "pucker"

Type	Values	Dimensions
Speech	MFCC + E	13
Speech	F0	1
Head Motion	Euler angles	3
Lips	Morph parameters	2

Table 3.2: Speech and motion features, with their respective dimensions.

and the distance between the top and bottom marker was mapped to "open". Both motion feature types were recorded at 500Hz and had to be down sampled to 200Hz using a low pass filter.

The reason why only four marker were used around the lips was the resolution of the motion capture system. Placing markers close together resulted in mixed up markers or merged markers. Additionally the head tracking was sometimes inaccurate because of occlusion. Furthermore there were inherent problems with the recognition of markers, which resulted in jittering. It is hoped that by employing a statistical model, the jitter problem can be overcome. A summary of all extracted features can be seen in Table 3.2.

Chapter 4

Hidden Markov Models

4.1 Hidden Markov Model Framework

4.1.1 Introduction

The dominant statistical models for both speech recognition and speech synthesis are Hidden Markov Models (HMMs) because of their efficient implementation and flexibility. In this thesis, HMMs are applied to motion capture data, and further used to generate animation parameter sequences. The following chapter explains the background theory of HMMs.

4.1.2 Model

A Hidden Markov Model is a probabilistic finite state machine where the only considered context is the the transition probability from the previous state to the current state. It is hidden because the observation is a probabilistic function of the state. The state is never observed but inferred from the observation.

The basic elements of an HMM are as follows:

- N is the number of states
- $A = \{a_{ij}\}$ describes the transition probability matrix, where a_{ij} is the probability

of making a transition from state i to state j, i.e.

$$a_{ij} = P(q_{t+1} = j | q_t = i) \tag{4.1}$$

- $B = \{b_j(\boldsymbol{o})\}$ is the set of emission probability distributions, where $b_j(\boldsymbol{o})$ is the probability distribution for for state j.
- $\pi = \{\pi_i\}$ are the prior state distributions where $\pi_i = P(q_o = i)$

The continuous output probability function $b_j(\boldsymbol{o})$ denotes a multivariate Gaussian mixture density function with M Gaussian functions, e.g.

$$b_j(\boldsymbol{o}) = \sum_{k=1}^{M} c_{jk} N\left(\boldsymbol{o}; \boldsymbol{\mu}_{jk}, \Sigma_{jk}\right) = \sum_{k=1}^{M} c_{jk} b_{jk}(\boldsymbol{o}) \tag{4.2}$$

where $N(\boldsymbol{o}; \boldsymbol{\mu}_{jk}, \Sigma_{jk})$ or $b_{jk}(\boldsymbol{o})$ denote a single Gaussian density function with mean vector $\boldsymbol{\mu}_{jk}$ and covariance matrix Σ_{jk} for state j and c_{jk} is the weight of the kth mixture component.

In compact notation the model can be written as:

$$\lambda = (A, B, \pi) \tag{4.3}$$

to describe the complete parameter set, which defines the probability measure $p(\boldsymbol{O}|\lambda)$.

4.1.3 The Three Basic Problems

Given the form of HMMs, Rabiner (Rabiner 1989) identified three basic problems:

- Given the observation sequence $\boldsymbol{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$ and a model λ, how do we efficiently compute $p(\boldsymbol{O}|\lambda)$, the probability density of the observation sequence, given the model?
- Given the observation sequence $\boldsymbol{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$ and a model λ, how do we choose a corresponding state sequence $\boldsymbol{q} = (q_1, q_2, \ldots, q_T)$, which is optimal in some meaningful sense?
- How do we adjust the model parameters λ to maximise $p(\boldsymbol{O}|\lambda)$?

The first problem can be described as given a model, what is the probability that an observed sequence was produced by that model. Looking at this problem from a recognition point of view, the problem can be seen as how well a model matches a given

observation sequence. If we have several models, the recognition point of view lets us choose the model that has the best correspondence to the observed sequence.

The second problem is about finding the hidden part of the model, the optimal unobserved state sequence. Since there is no single best sequence, this optimal sequence depends on the optimality criterion used, which in this case will be the most likely state sequence. Other optimality criterions are possible but will not be considered further here.

The final problem is the problem of training the HMM, which seeks to adjust the model parameters in such a way as to best describe a given observation sequence. This allows for the creation of models of existing experiences.

4.1.4 Problem 1: Probability Evaluation

Calculating the probability of the observation sequence, $\boldsymbol{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$ given the model λ can be done by enumerating every possible state sequence of length T. The state sequence

$$\boldsymbol{q} = (q_1, q_2, \ldots, q_T) \tag{4.4}$$

and the probability of an observation sequence $\boldsymbol{O}$, assuming statistical independence, this can be written as

$$p(\boldsymbol{O}|\boldsymbol{q}, \lambda) = \prod_{t=1}^{T} p(\boldsymbol{o}_t|q_t, \lambda) \tag{4.5}$$

$$= \prod_{t=1}^{T} b_{q_t}(\boldsymbol{o}_t). \tag{4.6}$$

The probability of the state sequence $\boldsymbol{q}$ given the model λ is

$$p(\boldsymbol{q}|\lambda) = \pi_{q_1} \prod_{t=2}^{T} a_{q_{t-1}q_t} \tag{4.7}$$

The probability that $\boldsymbol{O}$ and $\boldsymbol{q}$ happen at the same time, their joint probability is the product of the above two equations

$$p(\boldsymbol{O}, \boldsymbol{q}|\lambda) = p(\boldsymbol{O}|\boldsymbol{q}, \lambda) P(\boldsymbol{q}|\lambda). \tag{4.8}$$

The probability of $\boldsymbol{o}$ given λ is calculated by summing the joint probability over all possible state sequences $\boldsymbol{q}$

$$p(\boldsymbol{O}|\lambda) = \sum_{\text{all } \boldsymbol{q}} p(\boldsymbol{O}|\boldsymbol{q},\lambda)p(\boldsymbol{q}|\lambda) \tag{4.9}$$

$$= \sum_{q_1,q_2,\ldots,q_T} \pi_{q_1} b_{q_1}(\boldsymbol{o}_1) a_{q_1 q_2} b_{q_2}(\boldsymbol{o}_2) \ldots a_{q_{T-1} q_T} b_{q_T}(\boldsymbol{o}_T) \tag{4.10}$$

This direct evaluation of $p(\boldsymbol{O}|\lambda)$ is computationally very expensive as it involves $2T \cdot N^T$ multiplications. For every t there are N possible states and for each state sequence $2T$ calculations are needed for each term in the sum of Equation (4.10). Fortunately a more efficient algorithm exists, called the forward procedure.

4.1.4.1 Forward Algorithm

The forward variable $\alpha_t(i)$ is the probability of the partial observation sequence $\boldsymbol{o}_1 \boldsymbol{o}_2 \ldots \boldsymbol{o}_t$ and state i at time t, given the model λ. Using induction we can solve $p(\boldsymbol{O}|\lambda)$ in three steps

1. Initialisation

$$\alpha_1(i) = p([\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_t], q_t = i|\lambda) \qquad 1 \leq i \leq N \tag{4.11}$$

2. Induction

$$\alpha_{t+1}(j) = \left[\sum_{i=1}^{N} \alpha_t(i) a_{ij}\right] b_j(\boldsymbol{o}_{t+1}) \qquad 1 \leq t \leq T-1 \tag{4.12}$$

$$1 \leq j \leq N \tag{4.13}$$

3. Termination

$$p(\boldsymbol{O}|\lambda) = \sum_{i=1}^{N} \alpha_T(i) \tag{4.14}$$

Because the forward algorithm makes use of partially computed probabilities, its complexity is $O(N^2 T)$ rather than exponential. Similarly the backward algorithm calculates the probability of the partial observation sequence $\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_{T-1}, \boldsymbol{o}_T$ given that the HMM is in state i at time t, which is defined as

$$\beta_t(i) = p(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_{T-1}, \boldsymbol{o}_T|q_t = i, \lambda). \tag{4.15}$$

$\beta_t(i)$ can be calculated inductively;

1. Initialisation

$$\beta_T(i) = 1/N, \qquad 1 \leq i \leq N \tag{4.16}$$

2. Induction

$$\beta_t(i) = \sum_{j=1}^{N} a_{ij} b_j(\boldsymbol{o}_{t+1}) \beta_{t+1}(j) \qquad t = T-1, T-2, \ldots, 1 \tag{4.17}$$

$$1 \leq i \leq N \tag{4.18}$$

The relationship between between the two partial probabilities is such that α is computed recursively from left to right, and β is calculated recursively from right to left.

4.1.5 Problem 2: Decoding

Finding the optimal state sequence given an observation sequence is not straightforward as it depends on the definition of optimal. The optimality criterion used here is to choose the state q_t that is individually most likely at each time t. The a posteriori state occupation probability variable

$$\gamma_t(i) = p(q_t = i|\boldsymbol{O}, \lambda) \tag{4.19}$$

is defined as the probability of being in state i at time t, given the observation sequence $\boldsymbol{O}$ and the model λ. Using Bayes theorem $\gamma_t(i)$ can be expressed in this form

$$\gamma_t(i) = p(q_t = i|\boldsymbol{O}, \lambda) \tag{4.20}$$

$$= \frac{p(\boldsymbol{O}, q_t = i|\lambda)}{p(\boldsymbol{O}|\lambda)} \tag{4.21}$$

$$= \frac{p(\boldsymbol{O}, q_t = i|\lambda)}{\sum_{i=1}^{N} p(\boldsymbol{O}, q_t = i|\lambda)} \tag{4.22}$$

where $p(\boldsymbol{O}, q_t = i|\lambda)$ is equal to $\alpha_t(i)\beta_t(i)$. Therefore $\gamma_t(i)$ can be rewritten as

$$\gamma_t(i) = \frac{\alpha_t(i)\beta_t(i)}{\sum_{k=1}^{N} \alpha_t(k)\beta_t(k)} \tag{4.23}$$

Using Equation (4.23) the individually most likely state q_t^* at time t can be found using

$$q_t^* = \underset{i}{\operatorname{argmin}}\, \gamma_t(i) \tag{4.24}$$

However, determining the most likely state at every instant does not take the probability of state sequences into account. Therefore if some state transitions have zero probability the optimal state sequence is not guaranteed to be valid. By modifying the optimality criterion, and maximising $P(\boldsymbol{q}|\boldsymbol{O},\lambda)$ it is possible to solve for the single best state sequence.

4.1.5.1 Viterbi Algorithm

Dynamic programming can be utilised to find the best state sequence, $\boldsymbol{q} = (q_1, q_2, \ldots, q_T)$, given an observation sequence $\boldsymbol{O} = (\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_T)$. First the path with the highest score is defined as

$$\delta_t(i) = \max_{q_1,q_2,\cdots,q_{t-1}} p\left([q_1, q_2, \ldots, q_t = i], [\boldsymbol{o}_1, \boldsymbol{o}_2, \ldots, \boldsymbol{o}_t]|\lambda\right) \tag{4.25}$$

where $\delta_t(i)$ has the highest probability of any path that ends in state i at time t.

The complete algorithm includes an array $\psi_t(j)$ that keeps track of the argument that maximises

$$\delta_{t+1}(j) = \left[\max_i \psi_t(i)\, a_{ij}\right] b_j(\boldsymbol{o}_{t+1}) \tag{4.26}$$

at each t and j. The complete Viterbi algorithm is written as follow:

1. Initialisation

$$\delta_1(i) = \pi_i b_i(\boldsymbol{o}_1) \tag{4.27}$$

$$\psi_1(i) = 0 \tag{4.28}$$

2. Recursion

$$\delta_t(j) = \max_{1\le i\le N} \left[\delta_{t-1}(i)\, a_{ij}\right] b_j(\boldsymbol{o}_{t+1}) \tag{4.29}$$

$$\psi_t(j) = \operatorname*{argmax}_{1\le i\le N} \left[\delta_{t-1}(i)\, a_{ij}\right] \tag{4.30}$$

3. Termination

$$P^* = \max_{1\le i\le N} \left[\delta_T(i)\right] \tag{4.31}$$

$$q_T^* = \operatorname*{argmax}_{1\le i\le N} \left[\delta_T(i)\right] \tag{4.32}$$

4. Path backtracking

$$q_t^* = \psi_{t+1}(q_{t+1}^*) \tag{4.33}$$

The Viterbi algorithm is similar to the forward algorithm, but instead of summing partial probabilities from different paths coming to the same state, it picks and remembers the best path.

4.1.6 Problem 3: Training

The final problem identified by Rabiner is finding a method to adjust the model parameters (A, B, π) to maximise the probability of an observation sequence $\boldsymbol{O}$. Although there is no known method to analytically solve for an optimal parameter set, the likelihood $p(\boldsymbol{O}|\lambda)$ can be locally maximised using EM (expectation-maximisation) algorithm or Baum-Welch method as EM is known in the HMM context. Using the iterative EM method, $\xi_t(i,j)$ is defined as the probability of being in state i at time t, and state j at time $t+1$, given observation sequence $\boldsymbol{O}$ and the model λ, i.e.

$$\xi_t(i,j) = P(q_t = i, q_{t+1} = j|\boldsymbol{O}, \lambda). \tag{4.34}$$

By using the definitions of the forward ($\alpha_t(i)$) and backward ($\beta_t(i)$) variables $\xi_t(i,j)$ can be written as

$$\xi_t(i,j) = \frac{p(q_t = i, q_{t+1} = j, \boldsymbol{O}|\lambda)}{p(\boldsymbol{O}|\lambda)} \tag{4.35}$$

$$= \frac{\alpha_t(i)a_{ij}b_j(\boldsymbol{o}_{t+1})\beta_{t+1}(j)}{p(\boldsymbol{O}|\lambda)} \tag{4.36}$$

$$= \frac{\alpha(i)a_{ij}b_j(\boldsymbol{o}_{t+1})\beta_{t+1}(j)}{\sum_{k=1}^{N}\sum_{l=1}^{N}\alpha_t(i)a_{kl}b_l(\boldsymbol{o}_{t+1})\beta_{t+1}(j)}. \tag{4.37}$$

The previous definition of $\gamma_t(i)$ as the probability of being in state i at time t, given the observation sequence $\boldsymbol{O}$ and the model λ, can be related to $\xi_t(i,j)$ by summing over j, i.e.

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j). \tag{4.38}$$

Summing $\gamma_t(i)$ over the time index t, yields a quantity that can be interpreted as the expected number of times that state i is visited, or the expected number of transitions made from state i. Furthermore $\xi_t(i,j)$ can also be summed over t, which can be interpreted as the expected number of transitions from state i to state j.

By using the quantities $\gamma_t(i)$ and $\xi_t(i,j)$ the re-estimation formulas for π and A are

$$\bar{\pi}_j = \gamma_t(i) \tag{4.39}$$

$$\bar{a}_{ij} = \frac{\sum_{t=1}^{T-1} \xi_t(i,j)}{\sum_{t=1}^{T-1} \gamma_T(i)} \tag{4.40}$$

For continuous HMMs, where $b_j(\boldsymbol{o})$ is given as a mixture of Gaussian distributions (GMM) the re-estimation of the emissions probability $B = \{b_j(\boldsymbol{o})\}$ is computed with respect to the Gaussian parameters $c_{jk}, \boldsymbol{\mu}_{jk}$ and Σ_{jk} for the mixture model $b_j(\boldsymbol{o}) = \sum_{k=1}^{M} c_{jk} N(\boldsymbol{o}; \boldsymbol{\mu}_{jk}, \Sigma_{jk})$. The solutions are:

$$\bar{\boldsymbol{\mu}}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \boldsymbol{o}_t}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{4.41}$$

$$\bar{\Sigma}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k) \dot{(}\boldsymbol{o}_t - \bar{\boldsymbol{\mu}}_{jk})(\boldsymbol{o}_t - \bar{\boldsymbol{\mu}}_{jk})^\top}{\sum_{t=1}^{T} \gamma_t(j,k)} \tag{4.42}$$

$$\bar{c}_{jk} = \frac{\sum_{t=1}^{T} \gamma_t(j,k)}{\sum_{t=1}^{T} \sum_{k=1}^{M} \gamma_t(j,k)} \tag{4.43}$$

where $\gamma_t(j,k)$ is the component occupation probability, defined as the probability of being in the kth mixture component in state j at time t. e.g.

$$\gamma_t(j,k) = \frac{\sum_{i=1}^{N} \alpha_t(i) a_{ij} c_{jk} b_{jk}(\boldsymbol{o}_t) \beta_{t+1}(j)}{\sum_{i=1}^{N} \alpha_T(i)} \tag{4.44}$$

These formulas form the expectation step (E-step) and the computed expected values are used in the maximisation step (M-step) to compute the re-estimated model. The current model is defined as $\lambda(A,B,\pi)$, which is used to calculate the re-estimated model, defined as $\bar{\lambda} = (\bar{A}, \bar{B}, \bar{\pi})$.

The computation of $\bar{\lambda}$ can either result in $\bar{\lambda} = \lambda$, which means that a critical point in the likelihood function is reached, or that $\bar{\lambda}$ is more likely than λ. In other words, it is

more likely that the observation sequence has been produced by the new model, in the sense that $p(\boldsymbol{O}|\bar{\lambda}) > p(\boldsymbol{O}|\lambda)$.

By iteratively repeating the re-estimation procedure, and replacing λ with $\bar{\lambda}$, the likelihood that $\boldsymbol{B} = \boldsymbol{O}$ is being produced by the model is improved. Using a stopping criterion, the final result will be an maximum-likelihood (ML) estimate of the HMM. Alternatively, maximising Baum's auxiliary function

$$Q(\lambda, \bar{\lambda}) = \sum_{\boldsymbol{q}} p(\boldsymbol{q}|\boldsymbol{O}, \lambda) \log p(\boldsymbol{O}, \boldsymbol{q}|\bar{\lambda}) \tag{4.45}$$

over λ increases the likelihood, such that: $p(\boldsymbol{O}|\lambda) \leq p(\boldsymbol{O}|\bar{\lambda})$.

4.1.7 Parameter Tying

The more varieties or types in the data we try to model, the more training data is needed. Increasing the number of units, models, or context can quickly lead to data sparsity problems. In speech recognition a number of methods have been developed to address the sparsity problem.

In this thesis parameter tying by tree-based clustering (Young et al. 1995) is employed. States that have similar properties are tied together, meaning that they share parameters. Using decision tress with yes/no questions all the states for each context independent model are clustered. States sharing the same leaf nodes are tied. Using this technique even unseen contexts can be modelled.

4.2 Multimodal HMMs

The previous section introduced HMMs as a model for doing pattern recognition, particularly sequence recognition. These models usually model one type of data, which in the case of speech recognition would be the acoustic properties of the speech signal. In addition to modelling only one type of signal, HMMs can be used to model multimodal signals. These multimodal HMMs can perform stream mapping from one modality to the other. For example, given the audio signal, the HMMs produce the visual signal. Several different models based on HMMs have been developed by other

researchers that can do mapping between streams. The following sections gives a short review over the main previously developed techniques.

4.2.1 Input-output HMMs

Input-output HMMs (IOHMM) were introduced by Bengio & Frasconi (1995) for sequence processing. Li & Shum (2006) applied these models to the audio/visual mapping problem. In an IOHMM the emission and transition probabilities are conditional on the input sequence. Training these models is more complex than regular HMMs because the transition probability matrix is conditional on the input. Bengio et al. proposed to do the mapping from the input to the transition probabilities via neural networks. In addition, the emission probabilities can be specified using neural networks as well but its usually sufficient to use a Gaussian distribution. In Bengio's model the emission probability is given by

$$b_i(\boldsymbol{o}) = G(\boldsymbol{\mu}_i \varphi(\boldsymbol{o}_t), \sigma_i \varphi^2(\boldsymbol{o}_t)) \tag{4.46}$$

where $\boldsymbol{\mu}_i$ is the mean vector, σ_i is the co-variace matrix, and φ is the distance of the input signal to the origin. The transition probability matrix A is estimated using neural networks during the EM training of the model. Each entry in the transition matrix is given by an independent neural network after the M step.

4.2.2 Correlation HMM

Aleksic & Katsaggelos (2004) specified an extension of the HMM paradigm, called the correlation HMM, to do the mappping between acoustic and visual observation sequences. Each type observation is modelled by a separate HMM, the visual HMM λ^M and the audio HMM λ^S respectively, allowing for different topologies. The correlation HMM λ^C takes down-sampled acoustic observations $\boldsymbol{O}^S$ as input, and uses the Viterbi algorithm to generate a corresponding visual state sequence $\hat{\boldsymbol{q}}^M$, i.e.

$$\hat{\boldsymbol{q}}^M = \underset{\boldsymbol{q}^M}{\operatorname{argmax}} P(\boldsymbol{q}^M | \boldsymbol{O}^S, \lambda^c) \tag{4.47}$$

The topology of λ^C is the same as that of λ^M and their transition probabilities are the same ($A^C = A^M$). Only the emission probabilities are estimated for the CHMMs.

4.2.3 Regression mapping

Chen (2001) proposed a mapping method based on a least mean squared error (LMSE) optimality criterion, where a set of HMMs is trained on the joint audio-visual feature vector $\boldsymbol{o} = \left[\boldsymbol{o}^S, \boldsymbol{o}^M\right]$, where $\boldsymbol{o}^S$ is the audio vector and $\boldsymbol{o}^M$ is the visual vector. Since the co-variance matrix for each state distribution is diagonal, it is straightforward to extract an audio HMM from the joint HMM. Given the audio HMM and using the Viterbi algorithm, the optimal state sequence is found. Assuming that the optimal joint HMM sequence is the same as the optimal audio HMM one, the visual feature vector can estimated for each state. The optimal estimate $\hat{\boldsymbol{o}}^M$ in the LMSE is given by

$$\hat{\boldsymbol{o}}_t^M = E\left[\boldsymbol{o}_t^M | \boldsymbol{o}_t^S\right] \tag{4.48}$$

$$= \sum_i \frac{c_i N\left(\boldsymbol{o}_t^S; \boldsymbol{\mu}_i^S, \Sigma_i^S\right)}{f_{q_t}(\boldsymbol{o}_t^S)} \left(\boldsymbol{b}_i^\top \begin{bmatrix} 1 \\ \boldsymbol{o}_t^S \end{bmatrix}\right) \tag{4.49}$$

where c_i is the ith mixture coefficient with the audio distribution $N\left(\boldsymbol{o}_t^S; \boldsymbol{\mu}_i^S, \Sigma_i^S\right)$.

This method does not take the dynamics of the signal into account, and therefore has no guarantee of producing continuous output.

4.2.4 HMM-inversion

Choi et al. (2001) developed a method for estimating a visual feature vector from a known audio feature vector using a trained audio-visual HMM andNakamura (2002) proposed a similar method earlier based on EM. Audio and video are modelling jointly using M Gaussian mixtures with the emission probability for audio observations $\boldsymbol{o}^S$ and visual observations $\boldsymbol{o}^M$ for each state being

$$b_j(\boldsymbol{o}^S, \boldsymbol{o}^M) = \sum_{i=1}^{M} c_i N\left(\boldsymbol{o}^S, \boldsymbol{o}^M; \boldsymbol{\mu}^{SM}, \Sigma^{SM}\right) \tag{4.50}$$

where $\boldsymbol{\mu}^{SM}$ and Σ^{SM} denote the joint audio visual mean vector and covariance matrix respectively. The method uses a re-estimation method to find the optimal observation

sequence $\boldsymbol{o}^{\bar{M}}$ by maximising the auxiliary function

$$\bar{\boldsymbol{o}}^{M} = \underset{\hat{\boldsymbol{o}}^{M} \in S}{\operatorname{argmax}} Q\left(\lambda^{SM}, \lambda^{SM}; \boldsymbol{o}^{S}, \boldsymbol{o}^{M}, \hat{\boldsymbol{o}}^{M}\right). \tag{4.51}$$

where $\hat{\boldsymbol{o}}^{M}$ denotes the sequence of estimated visual parameters. Note that there are two identical λ^{SM} because the E-step does not estimate the model parameters but rather the video observation sequence $\hat{\boldsymbol{o}}^{M}$. Using EM to maximise the auxiliary function Q seems more robust than using Viterbi to estimate a state sequence first as it finds optimal estimates in the maximum likelihood sense. However the proposed method only considers the static component of the feature vector in the conversion and therefore does not guarantee continuous output.

4.2.5 Re-mapped Multimodal HMM

Brand & Shan (1998), Brand (1999), Brand & Hertzmann (2000) proposed an audio to visual mapping method based on the assumption that both data types can be represented with the same underlying modelling structure. The procedure starts by training HMMs on the video data. Then, for each HMM, the emission probabilities are mapped into the audio space during the M-step of the Baum-Welch training. Given a training observation sequence $\boldsymbol{O}$, the re-estimation formulas for the video and the audio HMM emission probability distribution parameters can be defined as follows

$$\gamma_t(i,j) = P(q_t = i, c = j|\boldsymbol{O}, \lambda) \tag{4.52}$$

$$\boldsymbol{\mu}_{ij}^{M} = \frac{\sum_{t=1}^{T} \gamma_t(i,j)\boldsymbol{o}_t^{M}}{\sum_{t=1}^{T} \gamma_t(i,j)} \tag{4.53}$$

$$\Sigma_{ij}^{M} = \frac{\sum_{t=1}^{T} \gamma_t(i,j)(\boldsymbol{o}_t^{M} - \boldsymbol{\mu}_{ij}^{M})(\boldsymbol{o}_t^{M} - \boldsymbol{\mu}_{ij}^{M})^{T}}{\sum_{t=1}^{T} \gamma_t(i,j)} \tag{4.54}$$

$$\boldsymbol{\mu}_{ij}^{S} = \frac{\sum_{t=1}^{T} \gamma_t(i,j)\boldsymbol{o}_t^{S}}{\sum_{t=1}^{T} \gamma_t(i,j)} \tag{4.55}$$

$$\Sigma_{ij}^{S} = \frac{\sum_{t=1}^{T} \gamma_t(i,j)(\boldsymbol{o}_t^{S} - \boldsymbol{\mu}_{ij}^{S})(\boldsymbol{o}_t^{S} - \boldsymbol{\mu}_{ij}^{S})^{T}}{\sum_{t=1}^{T} \gamma_t(i,j)} \tag{4.56}$$

Note that in re-estimation formulas for $\boldsymbol{\mu}$ and Σ for both audio and video, the a posteriori probability $\gamma_t(j,m)$ of emitting an observation from mixture component m at state j at time t is the same.

The mapping is done using the Viterbi algorithm, where given an input audio sequence and the trained audio HMM, the best state sequence $\hat{\boldsymbol{q}} = (q_1, q_2, \ldots, q_T)$ is obtained. Since for each audio state a corresponding state exists in the video HMM, it is straightforward to produce a mapping once the state sequence is known. Finally we can minimise the following formula to actually predict the visual trajectory $\boldsymbol{O}^*$.

$$\begin{aligned} \boldsymbol{O}^* &= \underset{O}{\operatorname{argmax}} \log \prod_t N(\boldsymbol{o}_t; \boldsymbol{\mu}_{q_t}, U_{q_t}) \\ &= \underset{O}{\operatorname{argmin}} \sum_{t=1}^{T} (\boldsymbol{o}_t - \boldsymbol{\mu}_{q_t})^T U_{q_t}^{-1} (\boldsymbol{o}_t - \boldsymbol{\mu}_{q_t}) \end{aligned} \tag{4.57}$$

where $\boldsymbol{o}_t = \left[\boldsymbol{o}_t^M, \Delta\boldsymbol{o}_t^M\right]$ and $\boldsymbol{\mu}_{q_t} = \left[\boldsymbol{\mu}_{q_t}^M, \boldsymbol{\mu}_{q_T}^{\Delta M}\right]$. Therefore the observation $\boldsymbol{o}_t$ includes both the position $\boldsymbol{o}_t^M$ and the velocity $\Delta\boldsymbol{o}_t^M$ of the video features.

The method proposed by Brand is able to produce smooth output by taking the first derivative of the feature vector into account. Although smooth output is guaranteed, the dependency between the static features and the dynamic features is not taken into account in the optimisation.

4.2.6 Summary of Multi-modal HMMs

The multi-modal models described in this section are all trained on parallel audio and visual data. Some models are trained on both data streams in parallel and others are trained on either data independently. What type of training is done depends on the type mapping that is done. Some mapping type really assume that each frame in the video corresponds to a frame in the speech, where as other mapping procedures are not as strict and employ for a higher level mapping between the two streams. Table 4.1 gives an overview of the previously described models in terms of the type of mapping between the data streams. Generally the mapping is usually done by finding an optimality criterion that can be optimised in terms of generating one sequence given the other. By assuming that both streams are parallel, transitions between consecutive frames in one stream can be translated to the other stream. In other words the state sequence of the input stream corresponds exactly to the state sequence of the output stream. Although they all use a sequence model of speech, the synchrony assumed between speech and motion is very tight, meaning that all models assume frame wise or state wise synchrony between speech and motion. This is sensible for lip motion because because

Type of HMM	Mapping property	Simultaneous audio-video	Smooth output	Synchrony
Input/Ouput	ANN to map input to output observations	Yes	No	Frame
Regression	LMSE criterion to maximise visual sequence	Yes	No	Frame
Inversion	EM criterion to maximise auxiliary function	Yes	No	Frame
Correlation	Viterbi employing Corr. HMM as cost function	No	No	State
Re-mapping	parallel HMMs sharing transition probabilities	No	Yes	State

Table 4.1: A comparison between the different multi-modal mapping approaches in terms of several factors. The mapping property refers to the type of optimisation or the type of model that is employed in the algorithm. Simultaneous audio-video refers to the type of training, if the initial models were trained on audio and visual information simultaneously or if the models were trained on audio and visual data separately. The output of the mapping procedure can either be continuous or discontinuous. The level of synchrony refers to either state-or frame-wise synchrony.

there is a direct correlation between the movement of the lips and sounds we make. For other types of motion like head motion where such a direct relationship is not apparent, other types of synchrony might work better.

4.3 HMM as a Signal Generator

4.3.1 Reformulation of an HMM as a Trajectory HMM

Tokuda et al. (2000), Zen et al. (2007) reformulated the HMM into the trajectory HMM to overcome specific shortcomings of the original model. First, the original HMM assumes a quasi static observation sequence since each part is represented by a state and intra-state time dependency cannot be accounted for. Second, the output probability

depends only on the current state, which is the state conditional independence assumption.

The original model is defined as a function of $\mathbf{o}$, where the static and the dynamic features are modelled as independent. The static feature vector with M dimensions is defined as

$$\boldsymbol{c}_t = [c_t(1), \ldots, c_t(M)]^\top \tag{4.58}$$

and the first and second order dynamic feature vectors, $\Delta \boldsymbol{c}_t$ and $\Delta^2 \boldsymbol{c}_t$, form

$$\boldsymbol{o}_t = \left[\boldsymbol{c}_t^\top, \Delta \boldsymbol{c}_t^\top, \Delta^2 \boldsymbol{c}_t^\top\right]^\top. \tag{4.59}$$

The sequences of the observations $\boldsymbol{o}$ and the static feature vectors $\boldsymbol{c}$ can be written in vector form as

$$\boldsymbol{o} = \left[\boldsymbol{o}_1^\top, \boldsymbol{o}_2^\top, \ldots, \boldsymbol{o}_T^\top\right]^\top \tag{4.60}$$

$$\boldsymbol{c} = \left[\boldsymbol{c}_1^\top, \boldsymbol{c}_2^\top, \ldots, \boldsymbol{c}_T^\top\right]^\top \tag{4.61}$$

$$\tag{4.62}$$

The explicit relationship between $\boldsymbol{o}$ and $\boldsymbol{c}$ can be written in matrix form:

$$\boldsymbol{o} = W\boldsymbol{c} \tag{4.63}$$

where W is a window matrix. Thus the value of a probability density function of a standard HMM λ is given by

$$p(\boldsymbol{o}|\lambda) = \sum_{\text{all } q} p(\boldsymbol{o}|\boldsymbol{q}, \lambda) P(\boldsymbol{q}|\lambda) \tag{4.64}$$

$$p(\boldsymbol{o}|\boldsymbol{q}, \lambda) = \prod_{t=1}^{T} p(\boldsymbol{o}_t|q_t, \lambda) \tag{4.65}$$

$$= \prod_{t=1}^{T} N(\boldsymbol{o}_t; \boldsymbol{\mu}_{q_t}, \Sigma_{q_t}) \tag{4.66}$$

$$= N(\boldsymbol{o}; \boldsymbol{\mu}_{\boldsymbol{q}}, \Sigma_{\boldsymbol{q}}). \tag{4.67}$$

$$\tag{4.68}$$

For simplicity it is assumed that each emission distribution is given as a single Gaussian distribution where $\boldsymbol{\mu}_{q_t}$ and Σ_{q_t} are the mean vector and covariance matrix respectively

at state q at time t. This formulation is improper because it allows inconsistencies between the static and dynamic feature vector sequences. The static and dynamic components are modelled as independent statistical variables.

Therefore a new statistical model in terms of $\boldsymbol{c}$ instead of $\boldsymbol{o}$ is defined as

$$p(\boldsymbol{c}|\lambda) = \sum_{\text{all } \boldsymbol{q}} p(\boldsymbol{c}, \boldsymbol{q}|\lambda) \tag{4.69}$$

$$= \sum_{\text{all } \boldsymbol{q}} p(\boldsymbol{c}|\boldsymbol{q}, \lambda) P(\boldsymbol{q}|\lambda) \tag{4.70}$$

The interdependencies between the static and dynamic part of the feature vector are explicitly modelled and therefore alleviate the deficiencies of the traditional HMM. So the output probability distribution of $\boldsymbol{c}$ conditioned on $\boldsymbol{q}$ is defined as

$$p(\boldsymbol{o}|\boldsymbol{q}, \lambda) = \frac{1}{Z_Q} N(W\boldsymbol{c}; \boldsymbol{\mu}_{\boldsymbol{q}}, \Sigma_{\boldsymbol{q}}) \tag{4.71}$$

$$= N(\boldsymbol{c}; \bar{\boldsymbol{c}}_{\boldsymbol{q}}, P_Q) \tag{4.72}$$

where $Z_{\boldsymbol{q}}$ is a normalisation term that validates the probability distribution. $\bar{\boldsymbol{c}}_{\boldsymbol{q}}, \boldsymbol{P}_{\boldsymbol{q}}$ and $\boldsymbol{r}_{\boldsymbol{q}}$ are the normalised parameters of the Gaussian distribution and computed from $\boldsymbol{W}, \boldsymbol{\mu}_{\boldsymbol{q}}$, and $\Sigma_{\boldsymbol{q}}$ as follows

$$\boldsymbol{R}_{\boldsymbol{q}} = \boldsymbol{W}^\top \Sigma_{\boldsymbol{q}}^{-1} \boldsymbol{W} = \boldsymbol{P}_{\boldsymbol{q}}^{-1} \tag{4.73}$$

$$\boldsymbol{r}_{\boldsymbol{q}} = \boldsymbol{W}^\top \Sigma_{\boldsymbol{q}}^{-1} \boldsymbol{\mu}_{\boldsymbol{q}} \tag{4.74}$$

$$\boldsymbol{R}_{\boldsymbol{q}} \bar{\boldsymbol{c}}_{\boldsymbol{q}} = \boldsymbol{r}_{\boldsymbol{q}} \tag{4.75}$$

Although this is the correct definition of the trajectory HMM and for precision these models should have been employed, the work conducted in this thesis only made use of the related smooth trajectory parameter generation algorithm described in the next section. It can be employed to synthesise smooth trajectories from standard HMMs, which do not take the explicit dependency between static and dynamic features into account during training. In this thesis this dependency is only taken into account in synthesis but not in training.

4.3.2 Parameter Generation

For synthesis a parameter sequence is generated from the HMM, by performing the unconstrained maximisation

$$\boldsymbol{o}_{max} = \underset{\boldsymbol{o}}{\operatorname{argmax}}\, p(\boldsymbol{o}|\boldsymbol{q},\lambda) \tag{4.76}$$

$$= \underset{\boldsymbol{o}}{\operatorname{argmax}}\, N(\boldsymbol{o};\boldsymbol{\mu}_{\boldsymbol{q}},\Sigma_{\boldsymbol{q}}). \tag{4.77}$$

Here we assume that the state sequence $\boldsymbol{q}$ is determined beforehand and $\boldsymbol{o}$ becomes equal to $\boldsymbol{\mu}_{\boldsymbol{q}}$, the speech parameter vector sequence becomes the sequence of mean vectors, which, results in a discontinuous output. Furthermore the values of the dynamic features in $\boldsymbol{o}$ are inconsistent with the values of the static features. The relationship between the static and the dynamic features should be modelled explicitly as a constraint of the maximisation Equation (4.77). By introducing a constraint, maximising $N(\boldsymbol{o};\boldsymbol{\mu}_{\boldsymbol{q}},\Sigma_{\boldsymbol{q}})$ with respect to $\boldsymbol{o}$ is equivalent to maximising with respect to $\boldsymbol{c}$, e.g.

$$\boldsymbol{c}_{max} = \underset{\boldsymbol{c}}{\operatorname{argmax}}\, N(\boldsymbol{W}\boldsymbol{c};\boldsymbol{\mu}_{\boldsymbol{q}},\Sigma_{\boldsymbol{q}}). \tag{4.78}$$

Since $\boldsymbol{c}_{max}$ is equivalent to the mean vector $\bar{\boldsymbol{c}}_{\boldsymbol{q}}$ of the trajectory HMM, employing the maximum likelihood criterion for estimating the model λ is similar to minimising the error between the training data $\boldsymbol{c}$ and the generated trajectory $\boldsymbol{c}_{max}$. In Figure 4.1 the state sequence given by each model is shown, where each state emits its mean vector, shown in the discontinous dotted line. Using the maximum likelihood parameter generation algorithm, a smooth trajectory is generated by taking the variance of each emission probability distribution into account, as well the dynamic constraints. The smaller the variance the closer the trajectory will be to the mean.

4.3.3 Global Variance

The statistical modelling of the signal with an HMM, smoothes the signal. In some cases the generated trajectories are devoid of many details, apparent in the originals, which can result in degradation of the final output. One of the details that seem to be missing, is the original variance, as the generated trajectories have a lower dynamic range than the originals. The maximum likelihood criterion in the parameter generation optimisation causes the trajectories to be close the mean vector sequence of the HMM, resulting in trajectories that display less variance.

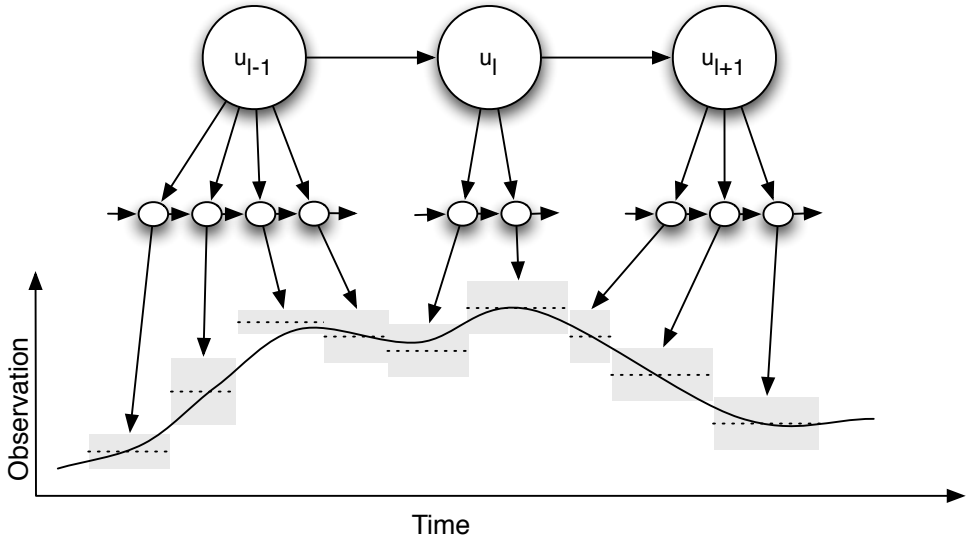

Figure 4.1: Maximum Likelihood Parameter Generation: The means of the emission probability distribution of each state is shown as the dotted line and the variance is shown as the grey bar. The smaller the variance the closer the trajectory will be to the mean.

Toda et al. (Toda & Tokuda 2007) developed a method to overcome this modelling deficiency, by implementing the global variance (GV) modelling within the HMM framework. The GV is the calculated as the following

$$\boldsymbol{v}(c) = [v(1), v(2), \ldots, v(d), \ldots, v(D)]^\top \tag{4.79}$$

$$v(d) = \frac{1}{T}\sum_{t=1}^{T}(c_t(d) - \bar{c}(d))^2 \tag{4.80}$$

$$\bar{c}(d) = \frac{1}{T}\sum_{\tau=1}^{T} c_\tau(d) \tag{4.81}$$

In the modified likelihood the GV is considered in addition to the static and dynamic feature vectors. The following likelihood is maximised

$$p(\boldsymbol{o}|\lambda, \lambda_v) = \sum_{\text{all } \boldsymbol{q}} p(\boldsymbol{o}, \boldsymbol{q}|\lambda)^{\omega} p(\boldsymbol{v}(c)|\lambda_v) \tag{4.82}$$

where $p(v(c)|\lambda_v)$ is modelled by a single Gaussian distribution with the mean vector

$\boldsymbol{\mu}_v$ and the covariance matrix Σ_v, e.g.

$$p(\boldsymbol{v}(c)|\lambda_v) = \frac{1}{\sqrt{(2\pi)^D|\Sigma_v|}} \exp\left(-\frac{1}{2}(\boldsymbol{v}(c)-\boldsymbol{\mu}_v)^\top \Sigma_v^{-1}(\boldsymbol{v}(c)-\boldsymbol{\mu}_v)\right) \tag{4.83}$$

The distribution λ_v and the HMM λ are trained independently and the constant ω controls the ratio between the two likelihoods.

The parameter generation algorithm basically operates with the additional constraint of the GV. It maximises the likelihood in Equation (4.82) with respect to $\boldsymbol{c}$. The likelihood $p(v(\boldsymbol{c}))$ can be viewed as a penalty term of the parameter generation because of the reduction of the GV. Using the EM algorithm $\boldsymbol{c}$ can be iteratively determined by maximising Equation (4.82).

Chapter 5

Formulation of Speech-driven Animation

5.1 Introduction

The ultimate goal of the research presented here is to animate a character given only the information present in a speech signal. In other words the parameters that govern the animation are determined by the speech input. One can describe this input/output relationship as a mapping from the speech parameter stream to the animation parameter stream. Although this mapping appears to be non-linear, by assuming a common origin or idea for both motion and speech, some dependency between these two streams follows. This dependency can be exploited by the proposed HMM-based machine learning approach. This chapter specifies the theory that underlies the system for synthesising motion from speech described in the thesis.

5.2 Multimodal Unit HMM

Previous multimodal HMMs described in Section 4.2 all present some solution to the problem of mapping from speech to motion. These previous approaches might give acceptable results for mapping between speech and lip motion, but there is little evidence for these models to work for other motion like movement of the head, which is less correlated with speech. In addition few details about the implementation of these

models are usually known. The model topology and the dependencies specified in previous work might only work for specific data sets but it is not always clear how general these approaches are.

Specifically regression mapping HMMs, described in Section 4.2.3 and EM-based HMM-inversion, described in Section 4.2.4, both use the same model topology and the same stream for both modalities. The dependencies between them are explicitly modelled in the covariance matrix. Although the algorithm that derives the visual output is different, both models utilise the same structure, that assumes a high degree of correlation between the visual and audio features. This makes it not straightforward to extend this type of model to other modalities.

Similarly, the Input-Output-HMM (IO-HMM) can be described as a straightforward graphical model that introduces dependencies between the input the output observations but the details of implementation remain elusive. Most implementations employ neural networks to model the dependencies, which can be difficult to train as the weights are not straightforward to optimise.

Given the problems of previous models, the lack of information in the literature and the specification of the current task, a new model type is introduced in this chapter, a multimodal unit HMM. It is multimodal because it can deal with more than one modality, for example speech and head motion, it uses motion units to produce a mapping between the modalities, and finally the underlying modelling framework is an HMM. This new model is general, in the sense that it can deal with many modalities, even ones that have very little correlation or only long range dependencies. It is also independent of topology, as it makes no assumptions about the structure of the HMM therefore making it possible to model different types of synchrony between streams.

5.3 Speech to Motion mapping

5.3.1 Definition of the Problem

The problem of mapping a speech signal to a motion signal can be explained for the case where we generate the motion vector sequences $\boldsymbol{O}^M = (\boldsymbol{o}_1^M, \boldsymbol{o}_2^M, \ldots, \boldsymbol{o}_T^M)$ from a given speech vector sequence $\boldsymbol{O}^S = (\boldsymbol{o}_1^S, \boldsymbol{o}_2^S, \ldots, \boldsymbol{o}_T^S)$ with a length of T frames. The

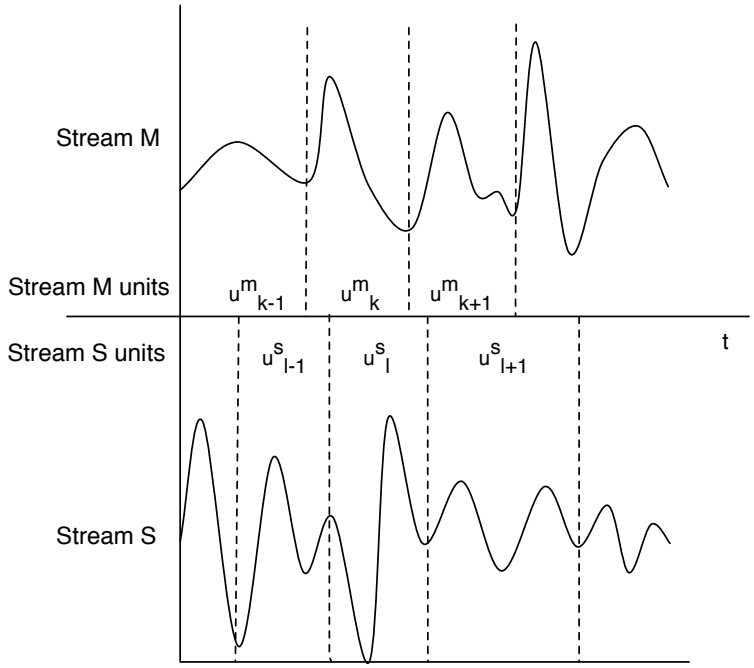

Figure 5.1: Two continuous data streams each marked with its own units.

lengths of the streams can differ, but for simplicity we assume the same length for both. Where, for example, o_i^M refers to a motion feature vector (e.g. Euler angles, in the case of head motion) and o_i^S to a speech feature vector (e.g. MFCCs). As we assume dependency between the speech and motion we can state the optimisation problem:

$$\boldsymbol{O}^{M*} = \underset{\boldsymbol{O}^M}{\operatorname{argmax}}\, p(\boldsymbol{O}^M, \boldsymbol{O}^S) \tag{5.1}$$

Similarily Equation (5.1) can be expressed as

$$\boldsymbol{O}^{M*} = \underset{\boldsymbol{O}^M}{\operatorname{argmax}}\, p(\boldsymbol{O}^M|\boldsymbol{O}^S)p(\boldsymbol{O}^S) \tag{5.2}$$

but the former expression of the problem will be considered. Although not apparent in Equation (5.1), both $\boldsymbol{O}^M$ and $\boldsymbol{O}^S$ are vector sequences of continuous values, making this problem quite different from speech recognition where the mapping is from a continuous stream to a stream of discrete symbols and no further. In the case of speech recognition, phonemes are the modelling unit, which also represent the recognised text and also the destination of the mapping. Stream to stream mapping on the other hand maps from one stream to another stream. Practically, stream to stream mapping can

be achieved by inserting a modelling unit layer between the two streams. Figure 5.1 shows two continuous streams, each marked with different units. It makes the problem of mapping from one continuous stream to another quite apparent as there is seemingly no clear relationship between the two. Introducing a unit layer as shown in Figure 5.1 can greatly reduce the complexity of the problem while maybe losing some information. It establishes a clearer and human readable relationship between two streams. Furthermore since the mapping between the two streams is complex and non-linear, the assumption can be made that more than one model will be needed for the mapping process.

Formally, the optimisation problem in Equation (5.1) can be simplified by incorporating the motion-unit sequence $\boldsymbol{u}^M = (u_1^M, ..., u_e^M)$, and the speech-unit sequence $\boldsymbol{u}^S = (u_1^S, ..., u_{e'}^S)$. Having defined modelling units for both streams, Equation (5.1) can be rewritten as an optimisation problem in terms of conditional probability, shown below.

$$\boldsymbol{O}^{M*} = \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}^M} \sum_{\boldsymbol{u}^S} p(\boldsymbol{O}^M, \boldsymbol{u}^M, \boldsymbol{u}^S, \boldsymbol{O}^S) \tag{5.3}$$

$$= \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}^M} \sum_{\boldsymbol{u}^S} p(\boldsymbol{O}^M | \boldsymbol{u}^M, \boldsymbol{u}^S, \boldsymbol{O}^S) p(\boldsymbol{u}^M, \boldsymbol{u}^S, \boldsymbol{O}^S) \tag{5.4}$$

$$= \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}^M} \sum_{\boldsymbol{u}^S} p(\boldsymbol{O}^M | \boldsymbol{u}^M, \boldsymbol{u}^S, \boldsymbol{O}^S) P(\boldsymbol{u}^M | \boldsymbol{u}^S, \boldsymbol{O}^S) p(\boldsymbol{O}^S | \boldsymbol{u}^S) | P(\boldsymbol{u}^S) \tag{5.5}$$

There are a variety of models that could be employed to produce the above probabilities. Dynamic Bayesian Networks (DBM) are a very popular sequence modelling family. In particular, Hidden Markov Models (HMM), which are one of the simplest, yet most powerful types of DBM, are widely used in the speech community. Figure 5.2 gives a graphical representation of an HMM where q_t represents the value of the hidden state variable at time t for a given data stream. To model the complete stream of data a string of HMMs are concatenated like in Figure 5.3. It illustrates that for each HMM there is a corresponding unit type at time t.

It is possible to simplify the computation further by assuming conditional independencies between the observation sequence.

$$\boldsymbol{O}^{M*} = \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}^M} p(\boldsymbol{O}^M | \boldsymbol{u}^M) \sum_{\boldsymbol{u}^S} P(\boldsymbol{u}^M | \boldsymbol{u}^S) p(\boldsymbol{O}^S | \boldsymbol{u}^S) P(\boldsymbol{u}^S) \tag{5.6}$$

where

$$p(\boldsymbol{O}^M | \boldsymbol{u}^M) = \sum_{\boldsymbol{q}^M} p(\boldsymbol{O}^M, \boldsymbol{q}^M | \boldsymbol{u}^M) \tag{5.7}$$

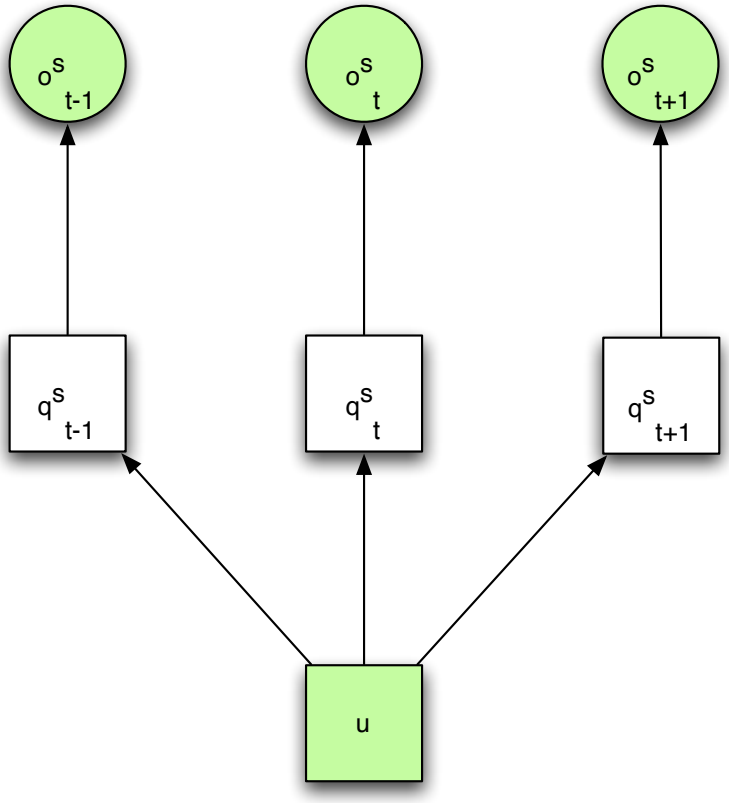

Figure 5.2: An single HMM of the unit u. The hidden state sequence Q is denoted by the state variable q_t at time t.

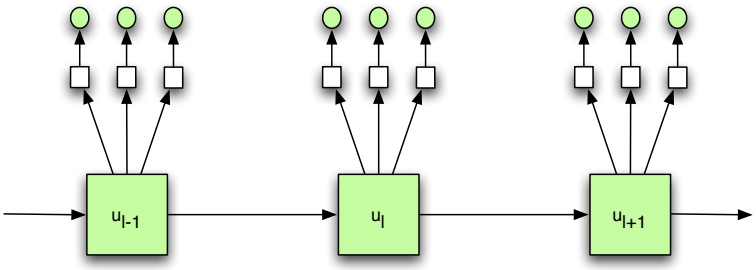

Figure 5.3: A string of HMMs given by the unit sequence and denoted by the variable u_l at time l

$$p(\boldsymbol{O}^S|\boldsymbol{u}^S) = \sum_{\boldsymbol{q}^S} p(\boldsymbol{O}^S, \boldsymbol{q}^S|\boldsymbol{u}^S) \tag{5.8}$$

Figure 5.4 shows a graphical representation of Equation (5.6), where each stream has a separate set of units with two HMMs modelling the speech and motion streams independently. Equation (5.8) is a probability where Equation (5.7) can be be estimated in a similar fashion to speech recognition using HMMs and statistical "language" models. The term $P(\boldsymbol{u}^M|\boldsymbol{u}^S)$ in Equation (5.6) is not straightforward as it maps from a discrete set of units to another discrete set. Furthermore it relates to the issue of synchrony between streams and their dependencies. If two different unit sequences are employed, the two streams could be asynchronous, where as if only one unit sequence is employed the two streams are synchronous.

Each unit is a discrete sequence of tokens that represent the corresponding continuous stream. By generating a unit sequence from one the streams it is possible to map between them. Generating a unit sequence is similar to speech recognition. Once the unit sequence is known, each unit has a trajectory model associated with it that can be used for generation.

5.3.2 Synchrony between streams

Synchronous models assume that all streams are derived from the same source of information. This is valid when data have been processed in different ways like in speech (RASTA, MFCC, LPC) but not necessarily valid when the data comes from different modalities. If the streams are synchronous, it is not clear if the synchrony is present on the state level, unit level, or even utterance level. In order to model streams from different modalities correctly, one has to at least address this problem.

It is also not clear how to express synchrony or asynchrony as both can happen on different levels, meaning there could be synchrony on the state level but not necessarily on the unit level and vice versa. Several possible alternatives on how to express synchrony between streams are presented here.

Figure 5.4 shows a graphical model that allows for state asynchrony between two streams. For each stream, a unit sequence is predicted as shown in the optimisation 5.6. The problem lies in determining the correspondence between sequences $\boldsymbol{u}^M$ and $\boldsymbol{u}^S$.

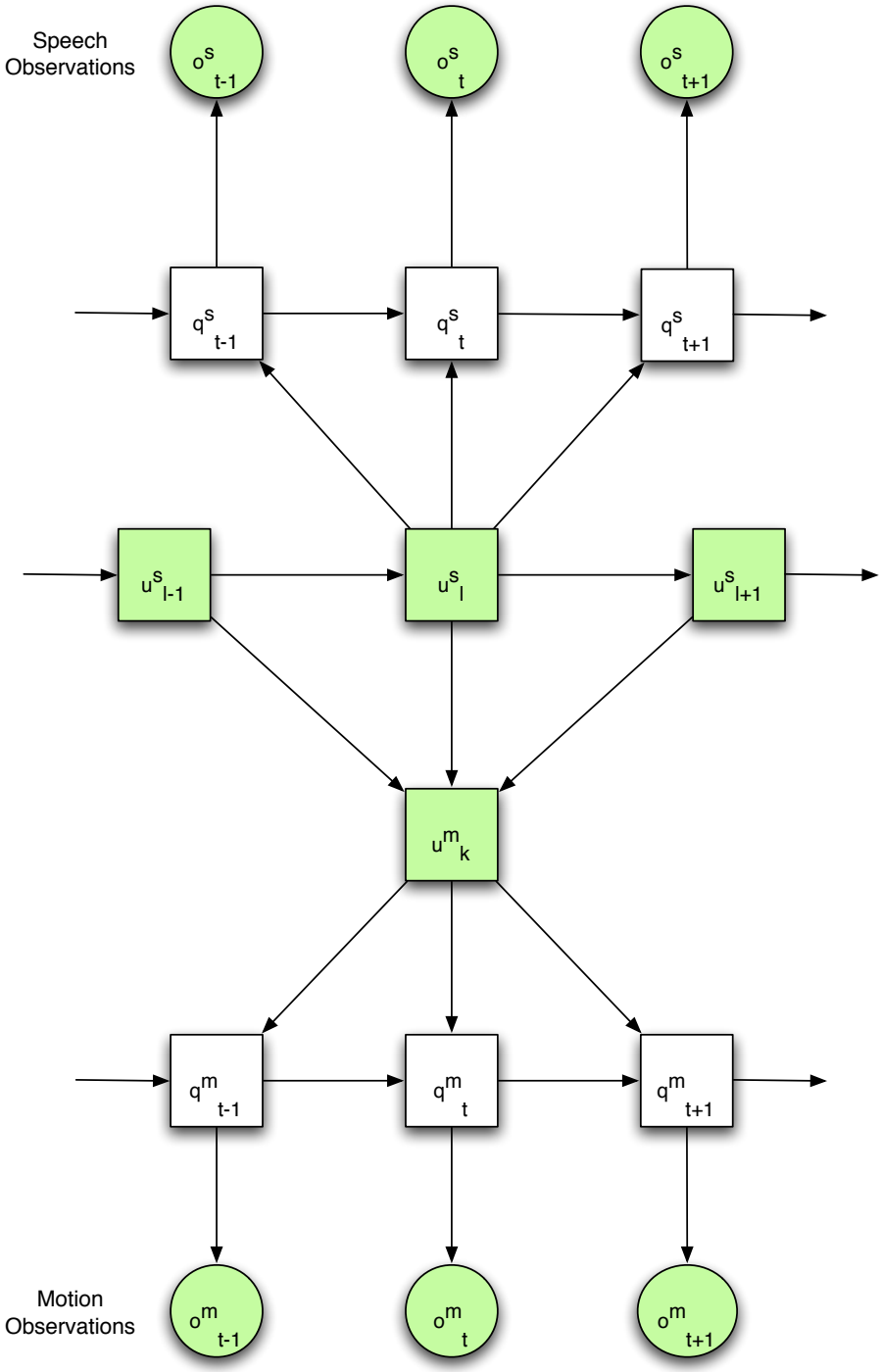

Figure 5.4: A graphical model representation of a generative model of speech and motion observations in two streams and two different units. q_t^s and q_t^m denote the hidden state variables at time, t, for speech and motion streams respectively. The speech unit u_S is mapped to the motion unit u_M.

It is possible to just employ one unit sequence $\boldsymbol{u}$ to produce a mapping between $\boldsymbol{O}^S$ and $\boldsymbol{O}^M$ although both streams have to be treated as synchronous or at least with a constant offset. This type of model is depicted in Figure 5.5 and expressed as the optimisation

$$\boldsymbol{O}^{M*} = \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}} p(\boldsymbol{O}^M|\boldsymbol{u},\boldsymbol{O}^S)p(\boldsymbol{O}^S|\boldsymbol{u})P(\boldsymbol{u}) \tag{5.9}$$

$$\simeq \underset{\boldsymbol{O}^M}{\operatorname{argmax}} \sum_{\boldsymbol{u}} p(\boldsymbol{O}^M|\boldsymbol{u})p(\boldsymbol{O}^S|\boldsymbol{u})P(\boldsymbol{u}) \tag{5.10}$$

$$\simeq \underset{\boldsymbol{O}^M}{\operatorname{argmax}}\, p(\boldsymbol{O}^M|\boldsymbol{u}^*) \tag{5.11}$$

where

$$\boldsymbol{u}^* = \underset{\boldsymbol{u}}{\operatorname{argmax}}\, p(\boldsymbol{O}^S|\boldsymbol{u})P(\boldsymbol{u}) \tag{5.12}$$

By assuming one common unit type for both streams it is not necessary anymore to treat them as separate. The problem can then be seen as single unit and single stream. Figure 5.6 expresses the problem as having one stream q and one unit type u with two observation sequences $\boldsymbol{O}^S$ and $\boldsymbol{O}^M$.

In summary two types of synchrony are expressed: unit level synchrony as shown in Figure 5.5 and state level synchrony as shown in Figure 5.6.

5.3.3 Dependency between Streams

In addition to synchrony, the level of dependency is another issue that needs to addressed in the modelling. Figure 5.6 shows a model were the speech observations and the motion observations are conditionally independent of each other. Depending on the type of features used there could be dependencies between them that are not explicitly modelled in this type of model. It is possible to introduce a dependency between the observations, as shown in Figure 5.7. However, it has been shown that in the case of speech synthesis introducing such explicit dependencies between articulatory observations and spectral observations does not improve the output quality (Ling et al. 2009) over a model without dependencies between observations. Additionally the introduction of these dependencies might mean a loss of generality as frame wise correlation is not always guaranteed.

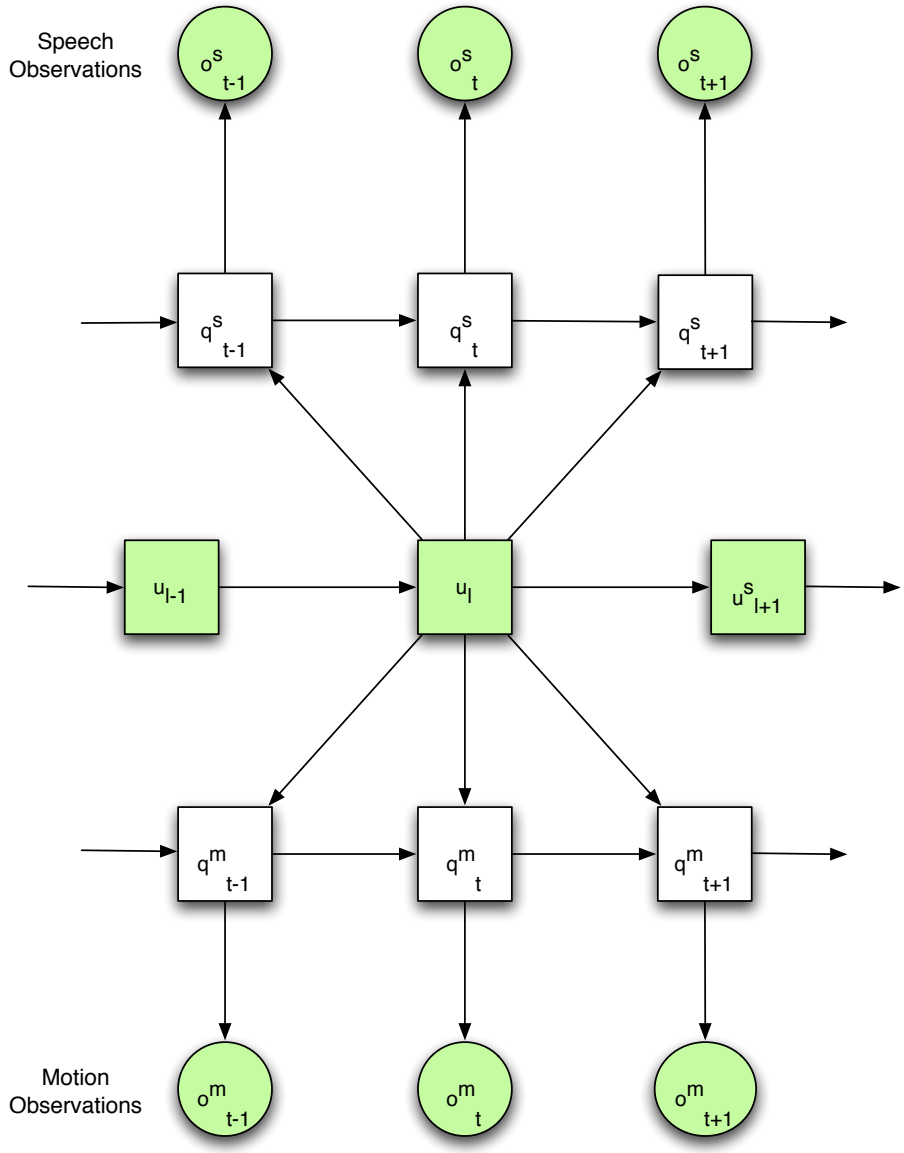

Figure 5.5: Unit-synchronous: A graphical model representation of a generative model of speech and motion observations in two streams and a single unit.

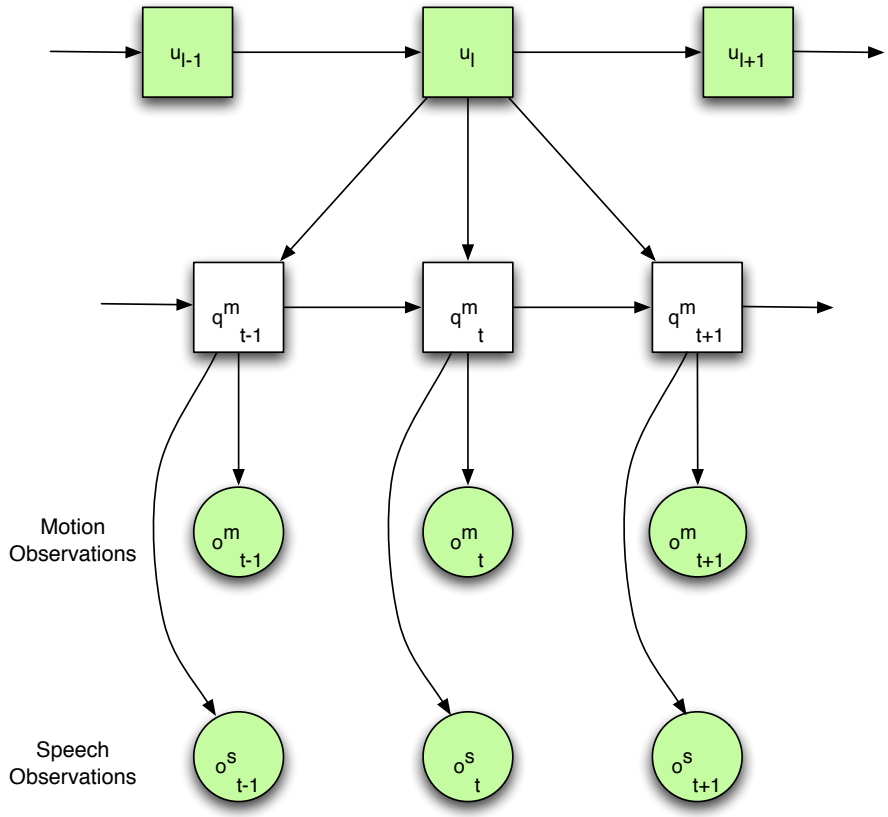

Figure 5.6: State-synchronous: A graphical model representation of a generative model of speech and motion observations in a single stream and a single unit.

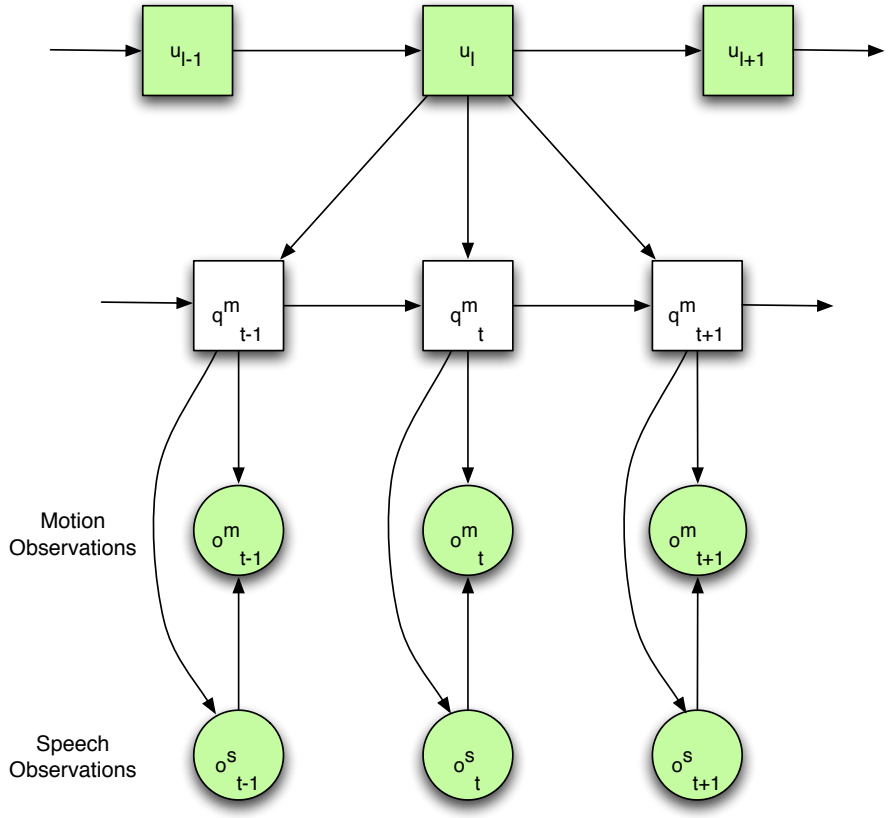

Figure 5.7: A graphical model representation of a generative model of speech and motion observations in a single stream and a single unit with dependencies between the two observations sequences.

5.3.4 Modelling Justification

Several different graphical models have been described in this chapter so far but the choice of model ultimately depends on the type of data used. Since the motion stream, in this thesis work will either consist of head motion features or lip motion features the correlation to the speech stream is not large. Additionally the synchrony between lip motion and speech and the synchrony between head motion and speech might be different. Therefore a more general modelling approach is needed.

The different modelling structures outlined in this chapter model the synchrony of streams on various levels but only two are considered in this thesis, unit level synchrony and state level synchrony. Additionally, the assumption is made that only one unit type is used in the modelling. The use of more than one type requires the development of a mapping method between the two unit sequences, which is out of scope. The choice of synchrony is evaluated in Chapter 7.

Furthermore, the dependency between observations needs to be taken into account. As there seems to be very little frame wise correlation between certain types of motion and speech, which has been shown in Chapter 7, introducing explicit dependencies between observations might not be beneficial. Therefore the assumption is made that the motion observations and the speech observations are conditionally independent, making the model more general in the sense that speech and motion are only dependent on unit level.

5.4 Motion Synthesis from a multimodal unit HMM

5.4.1 Determine Unit Sequence

Synthesis of motion is entirely separate from the modelling because once the unit sequence is determined, parameters will need to be generated from the corresponding models. So far most of the discussion has focused on modelling the various streams and mapping between them. Since we want to map from one stream to another, we have to determine the unit sequence $\boldsymbol{u}^*$ based on the speech observation sequence $\boldsymbol{O}^S$,

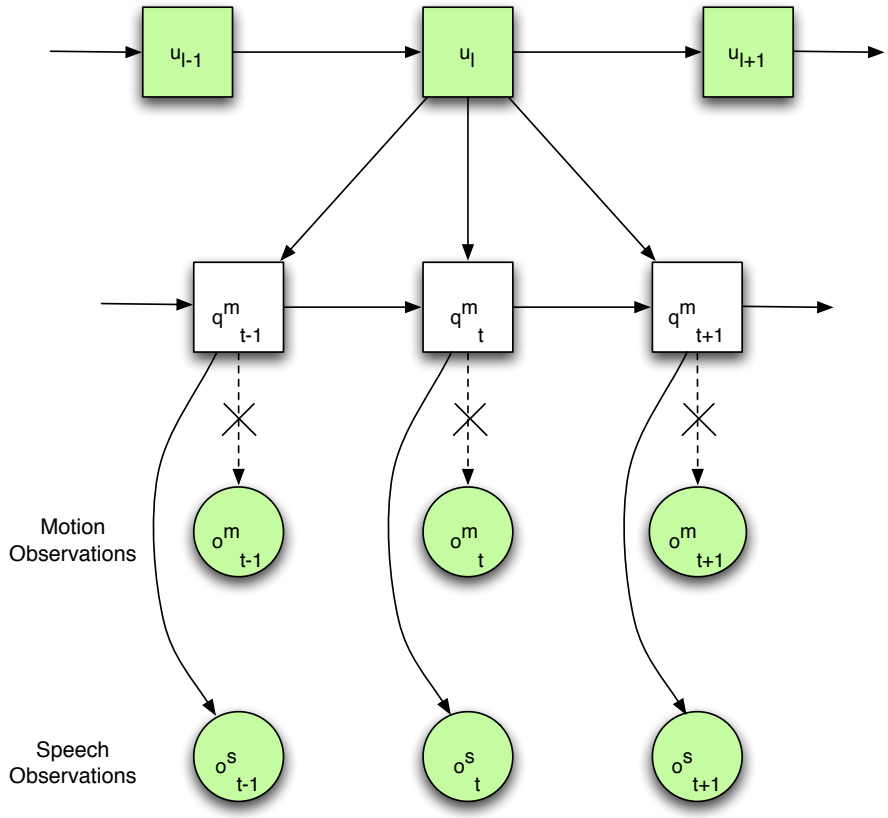

Figure 5.8: Only the speech observations are known and the unit sequence u_L is determined from it.

formally expressed as

$$\boldsymbol{u}^* = \underset{\boldsymbol{u}}{\operatorname{argmax}}\, p(\boldsymbol{O}^S|\boldsymbol{u})P(\boldsymbol{u}) \tag{5.13}$$

Figure 5.8 depicts the model where only the speech observations $\boldsymbol{O}^S$ are known. The motion observation sequence $\boldsymbol{O}^M$ is not used in the recognition.

5.4.2 Parameter Generation

Once the model sequence is known there needs to be a way to generate parameters from that sequence. In addition, the models only generate the motion observations as shown in Figure 5.9, by not considering the speech observations. Each state q_t of our HMMs consists of a mixture of Gaussian probability distributions but just using the mean or some other static value will give discontinuous output. An ideal algorithm

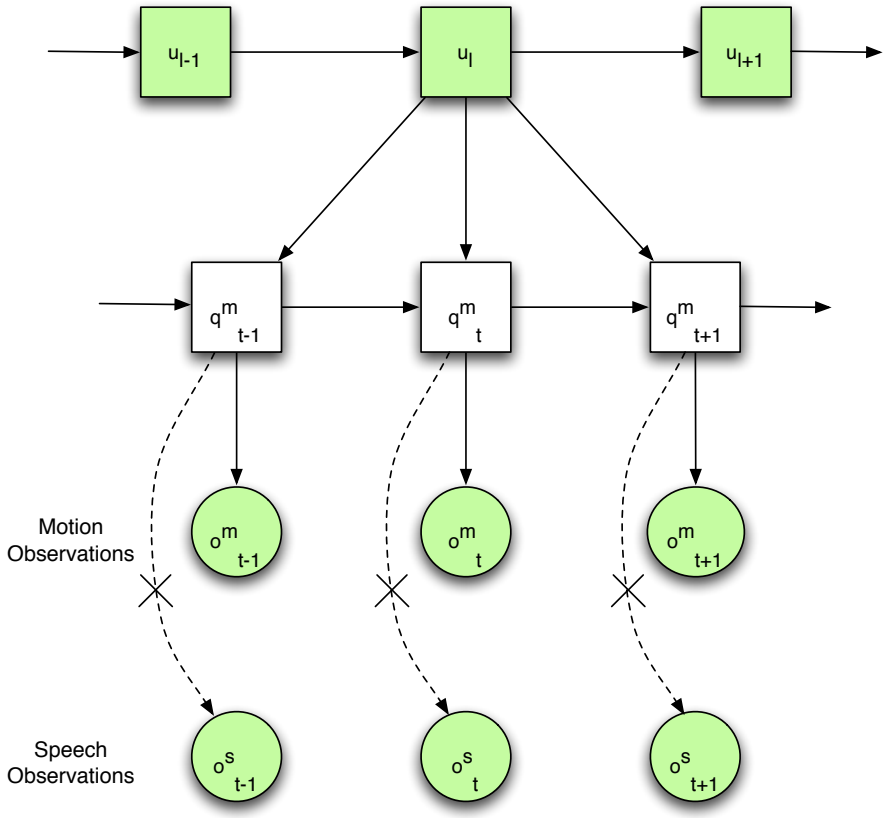

Figure 5.9: The motion observations are generated from the unit sequence u_L.

would produce smooth parameter trajectories that resemble the distribution of the original data. Chapter 5 gives details about the maximum likelihood parameter generation algorithm employed in this thesis. Furthermore, a method for controlling the dynamic range of the generated trajectories would be needed to have finer control over the output, which is described in Section 4.3.3. Finally, truly stochastic output of the model is necessary to achieve naturalness, as humans never behave in the exact same manner more than once and our models should inherently account for that.

5.4.2.1 Stochastic generation

One of the original contributions is the extension of the trajectory parameter generation algorithm to stochastic generation. In the deterministic parameter generation algorithm, the mixture that the parameter comes from is always the one with the highest

weight. The set of mixture components per state is defined as

$$b_{jk}(\boldsymbol{o}) = \sum_{i=1}^{M} c_i N(\boldsymbol{o}; \boldsymbol{\mu}_i, \Sigma_i). \tag{5.14}$$

During random generation, the vector $\mathscr{R} = (\mathscr{R}_1, \ldots, \mathscr{R}_M)$ is introduced where $\mathscr{R}_i = \delta(i-r)$ with r being a discreet uniform distribution in the range of $[1, M]$. e.g.

$$b_{jk}(\boldsymbol{o}) = \sum_{i=1}^{M} \mathscr{R}_i N(\boldsymbol{o}; \boldsymbol{\mu}_i, \Sigma_i). \tag{5.15}$$

The weights $\boldsymbol{c}$ are replaced by $\mathscr{R}$ so that during synthesis a mixture component is chosen randomly.

5.5 Multi-stream generation

An extension to the introduced paradigm can also be considered where mapping to more than one other modality is possible. For example given some speech, we generate both head motion and lip motion within the same modelling framework. Figure 5.10 shows joint lip, head, and speech model. We generate the motion vector sequences for lip and head movement, $\boldsymbol{O}^L = (\boldsymbol{o}_1^L, \boldsymbol{o}_2^L, \ldots, \boldsymbol{o}_T^L)$ and $\boldsymbol{O}^H = (\boldsymbol{o}_1^H, \boldsymbol{o}_2^H, \ldots, \boldsymbol{o}_T^H)$, from a given speech vector sequence $\boldsymbol{O}^S = (\boldsymbol{o}_1^S, \boldsymbol{o}_2^S, \ldots, \boldsymbol{o}_T^S)$ with a length of T frames. The optimal motion vector sequence $\boldsymbol{O}^{L^*}$ and $\boldsymbol{O}^{H^*}$ in the sense of probability would be obtained from the optimisation of a joint probability of $(\boldsymbol{O}^L, \boldsymbol{O}^H)$ given the sequence $\boldsymbol{O}^S$:

$$(\boldsymbol{O}^{H^*}, \boldsymbol{O}^{L^*}) = \underset{(\boldsymbol{O}^H, \boldsymbol{O}^L)}{\operatorname{argmax}}\, p(\boldsymbol{O}^H, \boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_H, \lambda_L) \tag{5.16}$$

where λ_L and λ_H are the sets of model parameters of the HMMs trained on lip and head movements, respectively. From the Bayes' theorem the joint probability can simply be written as

$$p(\boldsymbol{O}^H, \boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_H, \lambda_L) = p(\boldsymbol{O}^H | \boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_H) p(\boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_L). \tag{5.17}$$

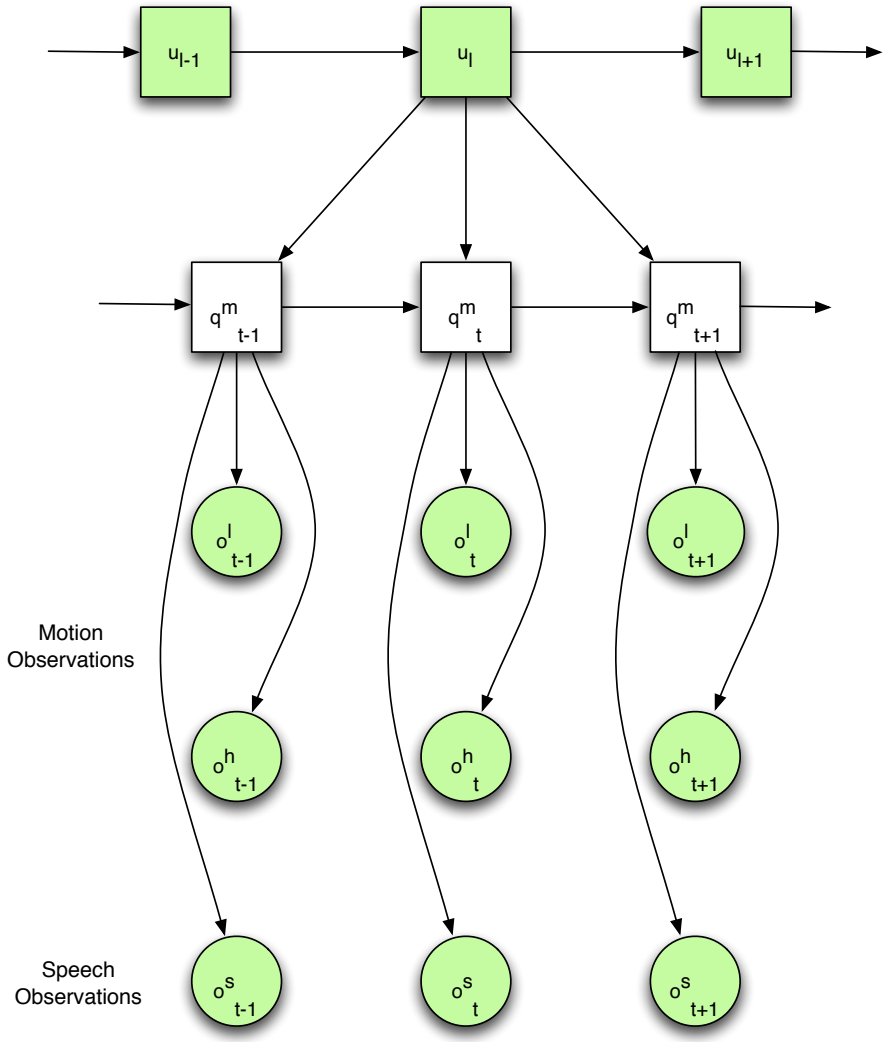

Figure 5.10: Multi-stream model. It models more than two modalities, two motion streams and one speech stream.

Moreover lip motion has a higher correlation with the speech vector sequence than does head motion. Thus we may solve the optimisation problem stepwise:

$$\boldsymbol{O}^{L^*} = \underset{\boldsymbol{O}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_L) \tag{5.18}$$

$$\boldsymbol{O}^{H^*} = \underset{\boldsymbol{O}^H}{\operatorname{argmax}}\, p(\boldsymbol{O}^H | \boldsymbol{O}^{L^*}, \boldsymbol{O}^S; \lambda_H). \tag{5.19}$$

We can work out the optimisation problems by incorporating the sets of motion-unit sequences $\boldsymbol{u}^L = (u_1^L, ..., u_e^L)$ and $\boldsymbol{u}^H = (u_1^H, ..., u_{e'}^H)$, which represent the lip and head movements corresponding to the given speech sequence. Using the motion labels units, the first optimisation regarding lip motion can be approximated by

$$\boldsymbol{O}^{L^*} = \underset{\boldsymbol{O}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^L, \boldsymbol{O}^S; \lambda_L) \tag{5.20}$$

$$= \underset{\boldsymbol{O}^L}{\operatorname{argmax}} \sum_{\forall \boldsymbol{u}^L} p(\boldsymbol{O}^L, \boldsymbol{u}^L, \boldsymbol{O}^S; \lambda_L) \tag{5.21}$$

$$\simeq \underset{\boldsymbol{O}^L, \boldsymbol{u}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^L | \boldsymbol{u}^L; \lambda_L) p(\boldsymbol{O}^S | \boldsymbol{u}^L) P(\boldsymbol{u}^L). \tag{5.22}$$

Thus we determine the lip motion unit sequence $\boldsymbol{u}^L$ from the given speech data $\boldsymbol{O}^S$ using the Viterbi algorithm and then generate a lip motion sequence from HMMs corresponding to the recognised units. For the probability $P(\boldsymbol{u}^L)$, we use back-off bi-gram models estimated from the training database. Once $\boldsymbol{O}^{L^*}$ is generated, we can simply apply the same procedures to the generation of head motion in the second optimisation:

$$\boldsymbol{O}^{H^*} \simeq \underset{\boldsymbol{O}^H, \boldsymbol{u}^H}{\operatorname{argmax}}\, p(\boldsymbol{O}^H | \boldsymbol{u}^H; \lambda_H) p(\boldsymbol{O}^S, \boldsymbol{O}^{L^*} | \boldsymbol{u}^H) P(\boldsymbol{u}^H). \tag{5.23}$$

Hence we recognise the head motion unit sequence $\boldsymbol{u}^H$ from the combined observation of the given speech data $\boldsymbol{O}^S$ and the generated lip motion $\boldsymbol{O}^{L^*}$ using the Viterbi algorithm, and generate a head motion sequence from the HMMs corresponding to the recognised units. Of course this algorithm could be used for additional movements such as eyebrows.

5.6 Conclusion

While this chapter described possible multi-stream models that are capable of modelling speech and motion simultaneously, the type of unit employed in the modelling

was not described. By introducing a unit layer to conduct the mapping between two streams, the choice of unit u_t becomes extremely important. Two properties of units in particular come to mind. First of all, what does the unit describe; is it speech or motion, or a mixture of both? Second, is the unit determined by the machine or should it be human readable, like phonemes?

In this thesis several choices of units are investigated, both speech-based units and motion-based units. A unit that is based on a stream would attach a particular label to parts of that stream, much like phonemes are the labels of speech but also a modelling unit. In Figure 5.5 the depicted modelling unit determines both the model of speech and motion. Choosing the unit u_t is delicate as it should represent information present in both streams. In particular, since we assume a common idea for both the motion and speech stream, the chosen unit needs to be describe motion but also bear a relationship to speech.

The choice of what the unit should describe is important but the meaning of the unit is also crucial. If one opts to let the unit be determined automatically, for example by clustering the data, then it might be more representative but it could also be meaningless. Trying to understand clusters is notoriously difficult, particularly when dealing with high dimensional data. On the other extreme are hand labelled units that explicitly attach meaning to segments in the data. These hand labels could be very helpful in application scenarios that involve human intervention at any of the synthesis stages, which is very common in computer animation since the artist wants to be in control of the final output. Furthermore it makes analysing the data more straightforward, as human-readable units provide the experimenter with feedback from the generated output, much like in speech synthesis where particular sounds are attached to phonemes.

For this research a manually defined unit that incorporates some knowledge about the data might be the most appropriate. The next chapter will describe experiments that were conducted on what type of unit is optimal in the context of the described models.

Chapter 6

Motion Analysis and Synthesis

6.1 Introduction

This chapter has three main themes: the analysis of the collected data, the investigation of the type of unit that is optimal for modelling motion, and the objective evaluation of the proposed models.

Experiments were conducted during the thesis based on the stream-to-stream mapping technique introduced in Chapter 5. Although the technique is general, it was first attempted to generate motion that has a high correlation to speech, i.e. lip motion. There has already been extensive research in the area of lip synchronisation and this thesis does not attempt to compete with state of the art systems. This attempt for lip synchronisation is a mere proof of concept that the developed mapping technique is reasonable. The next step was the synthesis of head motion because it is less related to speech than lip motion and a good application of the developed stream mapping process. Additionally, in the case of head motion, a detailed analysis of the collected data was carried out. Also as described in chapter 3, there is evidence that head motion in particular has more than a supportive function to the speech, in that it is actually part of the articulation. This chapter aims to add further evidence to this theory. This makes head motion synthesis the ideal application for the stream to stream mapping proposed in Chapter 5.

Finally, one of the most important aspects of the proposed method is the modelling unit, as it represents the heart of the mapping process. For all synthesis methods differ-

ent modelling units were investigated and a lot of effort was spent to find the optimal modelling unit for one particular type of motion.

6.2 Lip Motion Synthesis

Lip synchronisation is a difficult problem, which animators have been working on for a very long time. The aim of the proposed system is not to produce the best lip synchronisation but to be a first test of the proposed stream mapping procedure. The lip motion data was also used to do multi-stream-mapping as described in Section 5.5. Additionally it can provide lip synchronisation to the developed talking head, which is used in the perceptual evaluation. To demonstrate the described stream to stream mapping a short lip synchronisation experiment was conducted. The lip motion was modelled using a trajectory HMM for smooth parameter generation. The mapping method does not take frame-wise correlation between speech features and lip features into account, therefore the expected performance of the method in comparison to current techniques might be lower. The goal of the following experiments was not to beat the state-of-the-art but to demonstrate the feasibility of the method and to show the important role the modelling unit plays in the mapping.

6.2.1 Modelling Unit

For lip synchronisation with speech, visemes were used as the basic motion units. The data was phonetically labelled and a mapping between the phoneme labels and our viseme set was realised. To model the co-articulation we generated context-dependent units, meaning that for each viseme, there were different units depending on the left and right context. Furthermore, five different sets of visemes were implemented to test for the effects of different motion units. In particular the sets are described as follows: The 2vis set consists of just two visemes, one for open mouth, and one for closed mouth. The simple set (sVis) groups phonemes into 6 different classes that correspond roughly to the following phoneme classes: vowels are classed according to their height and backness, resulting in three classes and consonants are either bilabial, labiodental, or just plain consonants. The extended set (eVis), breaks these classes further down, distinguishing diphthongs (5), classifying more vowels on their own

Name	no of Visemes	Description
2vis	2	only 2 mouth shapes, open or close
pbVis	9	Preston Blair set, used in Disney animation
sVis	6	simple grouping of phonemes according to vowels and consonants
eVis	19	extending grouping, with diphthongs and vowel classes
phone	46	CMU phone set

Table 6.1: Viseme sets

(4), and making further distinctions between consonants (10), resulting in a total of 19 classes. Table 6.1 shows a comparison of the different sets. The actual mapping between phonemes and visemes is shown in Appendix A.

Although most lip-synchronisation methods utilise some viseme representation there has been no work that I have been aware of, comparing different viseme sets. Since the stream-mapping algorithm proposed in Chapter 5 is highly dependent on the modelling unit, it was decided to do a comparative study between unit types or viseme sets in the context of stream mapping.

6.2.2 Modelling Lip Motion

The proposed method models lip motion directly using the recorded motion capture points. The speech and motion data are simultaneously modelled using context-dependent HMMs. The data is described as a sequence of context dependent viseme models. Each model consists of five streams. One stream for the speech features, three streams for F0, and one stream for the motion features. The lip motion is modelled using two features, that is, the distance between the upper and lower lip and the distance between the left and right corner of the mouth. Additionally, the first and second derivative of the lip motion features are used to better model the dynamics of the motion trajectories. The speech is modelled using the first 12 mel cepstrum coefficients and energy. The first and second derivative of the speech features are also modelled. F0 is modelled using three streams, one for the static features and one stream each for the first and second derivative respectively. The HMMs use a mixture of Gaussian distributions at each state.

6.2.3 Synthesising Lip Motion

To synthesise lip motion, speech is the only required input to the model. Recognition is performed using the multi-stream HMMs. During the recognition step, only the speech feature streams are used, producing a sequence of visemes. The visemes give the sequence of context dependent models that are used for synthesising the actual trajectories.

Synthesising from a stochastic model like a conventional HMM is like rolling a dice. At each state, a value is sampled from the distribution, the resulting output is stochastic and discontinuous. Conventional HMMs are good at recognising patterns but the sampled trajectories are not representative of the actual trajectories that are in the training data. Using the parameter generation of the trajectory HMM, a value is not randomly sampled from each distribution but the ML estimation is performed on a string of distributions taking the first and second derivatives of the observations into account. Using this explicit constraint smooth output can be produced.

6.2.3.1 Optimal Motion

It is straightforward to justify the above procedures. For simplicity, let us explain the case where we generate motion vector sequences for the lips $\boldsymbol{O}^L = (\boldsymbol{o}_1^L, \boldsymbol{o}_2^L, \ldots, \boldsymbol{o}_T^L)$ from a given speech vector sequence $\boldsymbol{O}^S = (\boldsymbol{o}_1^S, \boldsymbol{o}_2^S, \ldots, \boldsymbol{o}_T^S)$ with a length of T frames. Lip motion has a high correlation with the speech vector sequence.. Thus we may solve the optimisation problem:

$$\hat{\boldsymbol{O}}^L = \underset{\boldsymbol{O}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^L|\boldsymbol{O}^S) \tag{6.1}$$

We can solve the optimisation problem by incorporating the motion-unit sequence $\boldsymbol{u}^L = (u_1^L, ..., u_e^L)$, which represent the lip movements corresponding to the given speech sequence. Using the motion label units, the first optimisation regarding lip motion can be approximated by

$$\begin{aligned}
\hat{\boldsymbol{O}}^L &= \underset{\boldsymbol{O}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^L|\boldsymbol{O}^S) && (6.2)\\
&= \underset{\boldsymbol{O}^L}{\operatorname{argmax}} \sum_{\boldsymbol{u}^L} p(\boldsymbol{O}^L|\boldsymbol{u}^L, \boldsymbol{O}^S) p(\boldsymbol{O}^S|\boldsymbol{u}^L) p(\boldsymbol{u}^L) && (6.3)\\
&\simeq \underset{\boldsymbol{O}^L}{\operatorname{argmax}}\, P(\boldsymbol{O}^L|\hat{\boldsymbol{u}}^L) && (6.4)
\end{aligned}$$

where

$$\hat{\boldsymbol{u}^L} = \underset{\boldsymbol{u}^L}{\operatorname{argmax}}\, p(\boldsymbol{O}^S|\boldsymbol{u}^L)p(\boldsymbol{u}^L) \tag{6.5}$$

Thus we recognise the lip motion units $\boldsymbol{u}^L$ from the given speech data $\boldsymbol{O}^S$ using the Viterbi algorithm and then generate a lip motion sequence from HMMs corresponding to the recognised units. For the probability $p(\boldsymbol{u}^L)$, we use back-off bi-gram models estimated from the training database.

6.2.4 Evaluation

6.2.4.1 Objective evaluation

An experiment was carried out to find the optimal unit for our model by comparing how well the lip closing in the synthesised data lined up with the original data. Lip closing here means when the distance of the upper and lower lip marker was within 20% of the smallest distance in the data. If a mouth closing occurred within 2 frames of the original mouth closing, a point was awarded. The test only measures recall but not precision. It does not penalise greedy synthesis that produces too many mouth closings. Still the test gives and indication of which viseme set might be better. The best performing sets were the extended version of a simple set designed by us (eVIS) and the Preston-Blair phoneme set (pbVis), which seems to be a standard in animation (Martin 2006). Table 6.2 shows the different sets and its score in our evaluation. For our data and model, the eVIS set yielded the best results. Example animations were also produced for the viseme sets and again the eVis set produced the highest quality animation as can be seen Figure 6.2.

What is interesting to note is that the Preston-Blair set seems to perform worse in our experiments than our own designed sets. This does not mean that one set is superior or inferior to another but that for automatic generation of lip animation, the viseme set used makes a difference. It is important, when modelling lip motion, to chose the viseme set carefully. Figure 6.1 shows a comparison between a synthesised lip motion trajectory and the original trajectory. The original movement has a higher dynamic range, which is a common problem when using stochastic modelling but otherwise the synthesised trajectory follows the original relatively closely. Still all scores were very

Name	no of Visemes	Mean Score of alignment
2vis	2	66.5
pbVis	9	67.7
sVis	6	67.4
eVis	19	**68.8**
phone	46	65.2

Table 6.2: Viseme sets and their scores. Better alignment between the mouth closings of the original utterance and the synthesised one produces a higher score.

close, which makes the interpretation of the results difficult. Therefore a perceptual evaluation was needed to further evaluate the viseme sets.

6.2.4.2 Perceptual Evaluation

To investigate the merit of the proposed method further, we conducted a perceptual evaluation. Six different utterances were synthesised in three variants, using the eVis viseme set, the full phoneme set, and the original tracking data. Ten speech technology experts were asked to judge the lip-synchronisation of our character comparing these three conditions. The instructions were given verbally with the experimenter close-by if any questions arise during the evaluation. The specific instructions were to identify to chose the preferred video. The participants saw two videos in succession and had to decide which one had better lip synchronisation. They could view each video as often as they like. Each permutation of the 3 conditions was seen twice by each participant to check for consistency giving a total of 36 trials. They were presented in randomised order. Figure 6.2 shows the preference score for each condition.

It is interesting to note that the specifically-designed viseme set is judged better than a standard phone set, when they are compared with each other. When both sets are directly compared with the original data, both sets are judged about equally worse than the original.

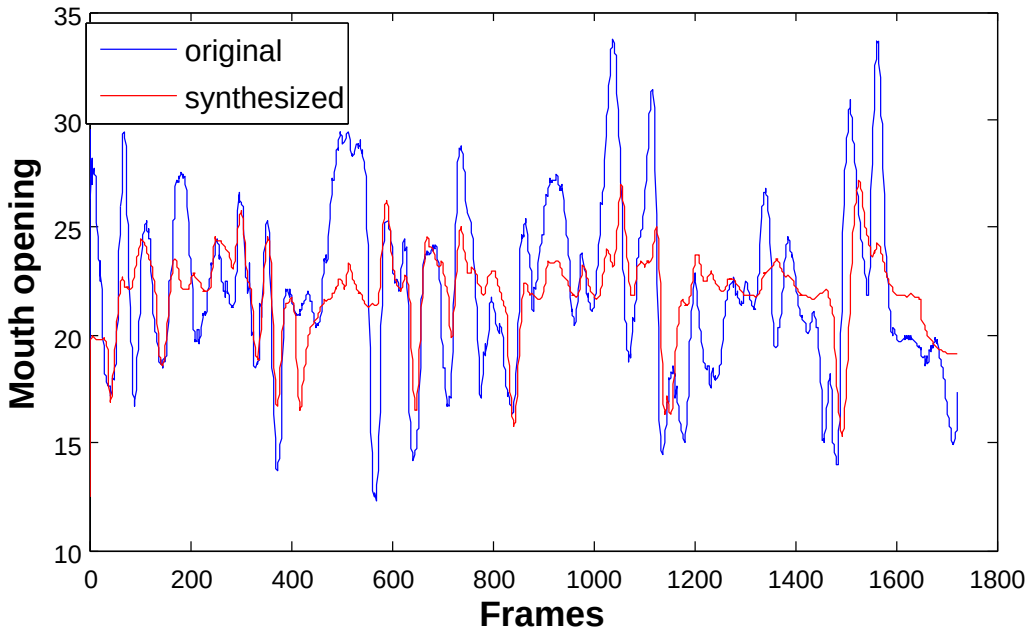

Figure 6.1: This figure shows a comparison of two trajectories. The synthesised trajectory clearly follows most of the original trajectory. The differences in the dynamic range are due to the nature of stochastic modelling.

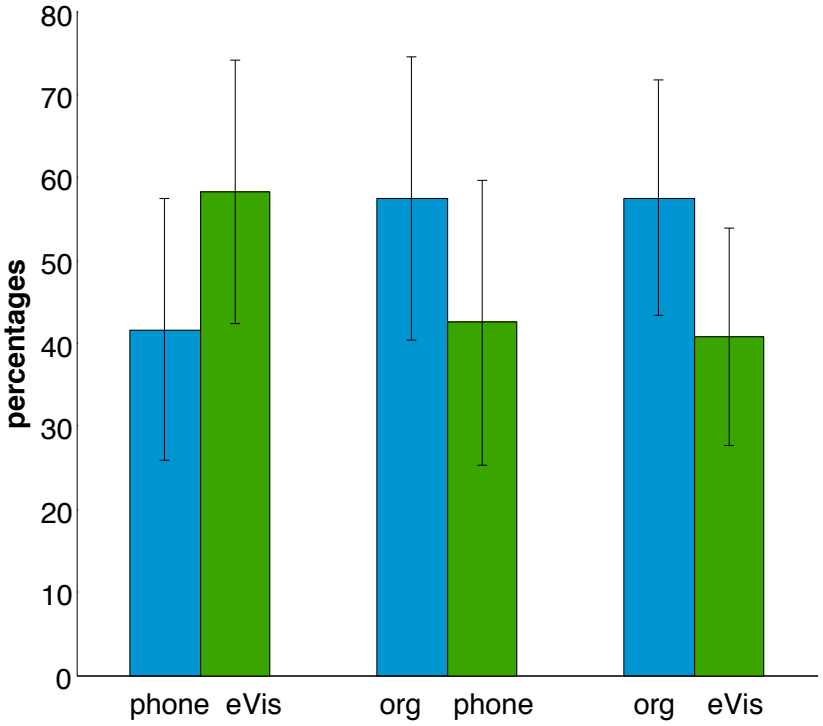

Figure 6.2: The bars show the ratio of preferences for each condition. For example 60 % preferred the eVis set over 40 % preferred the phone set. The original movement was rated best compared to the phoneme based and the viseme based movement. But the viseme based movement was rated higher than the phoneme based movement, when compared directly.

6.2.5 Summary & Discussion

This section demonstrated the feasibility of the proposed stream mapping method. As a first test of the proposed HMM-based method for synthesising motion, the lip motion generation was a simple task. The correlation between the lips and the speech is quite large. Interestingly different modelling units yield different results. Using modelling units that describe the motion rather than the speech, resulted in more favourable ratings. This is encouraging because it gives further evidence to notion that we should be modelling the motion rather than the speech, when doing speech driven animation. One of the major drawbacks of our system is the corpus we are using. Because of technical limitations we were only able to track 4 points around the mouth, which resulted in impoverished models. Theoretically, our models can work with an unlimited number of tracking points, even producing other types of animation than just lip motion.

Although the stream mapping method works to a certain extent, generating motion other than lip motion is the real test of the motion. However, the stream mapping depends on the modelling and for lip synchronisation the choice of modelling unit is currently confined to using phoneme or viseme based units, but for other types of motion the choice of unit is not as clear cut. Therefore the next sections will describe the process of modelling head motion with various unit types.

6.3 Head Motion Synthesis

Head motion is less related to speech than lip movement, but still highly synchronised. From observing people, one can see that the whole body moves in synchrony to underpin and support the message that we are trying to get across, head motion is no exception (McNeill 2005).

The stream to stream mapping is highly dependent on the modelling unit u and therefore a lot of effort was put into finding the best unit for each particular problem. In the case of head motion, since there is very little frame wise correlation between speech and motion, the space of possible units is larger than for the lip synchronisation. The changes in head motion happen slower than in speech, making it possible to employ longer units that span over a hundred frames. However, two broad categories of units are investigated in this thesis: speech based units and motion based units. Speech-

Unit type	Features	Label	Human-readable
speech	text	phoneme	yes
	acoustic	syllable	yes
		word	yes
		phrase	yes
motion	Euler angles	manual	yes
		automatic	no

Table 6.3: Comparison of speech-based and motion based unit types in terms of features and human readability.

based units, are for example phonemes where the unit has a direct correspondence to the speech signal, which tend to be shorter. Motion-based units on the other hand describe the motion signal, like a head shake in the case of head motion, which tend to be longer. Table 6.3 gives an overview of some of the possible candidates for the modelling unit. In addition, the issue of human understandable units needs to be addressed because the output of any automatic system will not be perfect and therefore will need to be edited to produce high quality animation.

Finally, the argument for non-deterministic output becomes more important, the further removed the synthesis becomes from the lips. With head motion, humans clearly move their head in many different ways, eventhough expressing and articulating the same concept. It is therefore important to account for this variability in the synthesis process, and one of the contributions of this thesis will be to evaluate if it contributes to the perceived naturalness of an animation. Besides the stochastic nature of the head motion, the dynamic range is another important factor that needs to be considered in the synthesis. Clearly, different intensities of the same type motion exist and need to be accounted for. Crucially, this non-deterministic dynamic range controllable output needs to be smooth, which in this case refers to continuous output. The final goal after all is still to produce a natural animation and the most important property for that is smoothness, in the sense that human movement is continuous.

6.4 Statistical Analysis

Various researchers have suggested a close relationship between speech and head motion; even frame-wise correlations have been suggested. Yehia et al. (2002) found correlations between F0 and head motion within utterances but could not find any globally. A linear model could predict head motion from F0 for a specific utterance if trained on that particular utterance. A model trained on all utterances failed to predict the correct head motion. To investigate the correlations in the collected data, frame wise correlations on our feature vector for all utterances and for each utterance individually were calculated. No substantial correlations within utterances or globally were found. Figure 6.3 shows a matrix plot of the correlations between F0, RMS energy, and Euler angles, with their respective derivatives for a particular utterance. The plot shows very clearly that there is no strong correlation among the different features contrary to some other findings Busso et al. (2006), Yehia et al. (2002).

Canonical Correlation Analysis To further investigate the absence of correlations, we used a more sophisticated technique to calculate correlations. Canonical Correlation Analysis (CCA) makes it possible to calculate correlations between vectors of different dimensions. It measures the linear relationship between two multidimensional variables by calculating an optimal base for each variable in respect to correlations. The dimensionally of the new bases is equal or less than the smallest dimensionally of the two variables. Formally (Borga 2001) CCA is defined as finding two sets of basis vectors, one for x and one for y, such that the correlations between the projections of the variables onto these basis vectors are mutually maximised. The linear combinations $x = \boldsymbol{x}^\tau \hat{\boldsymbol{w}}_x$ and $y = \boldsymbol{y}^\tau \hat{\boldsymbol{w}}_y$ respectively are used in the maximisation of the following function:

$$\rho = \frac{E[xy]}{\sqrt{E[x^2]E[y^2]}} \tag{6.6}$$

$$= \frac{E[\hat{\boldsymbol{w}}_x^\tau \boldsymbol{x}\boldsymbol{y}^\tau \hat{\boldsymbol{w}}_y]}{E[\hat{\boldsymbol{w}}_x^\tau \boldsymbol{x}\boldsymbol{x}^\tau \hat{\boldsymbol{w}}_x]E[\hat{\boldsymbol{w}}_y^\tau \boldsymbol{y}\boldsymbol{y}^\tau \hat{\boldsymbol{w}}_y]} \tag{6.7}$$

$$= \frac{\boldsymbol{w}_x^T \boldsymbol{C}_{xy} \boldsymbol{w}_y}{\sqrt{\boldsymbol{w}_x^\tau \boldsymbol{C}_{xx} \boldsymbol{w}_x \boldsymbol{w}_y^\tau \boldsymbol{C}_{yy} \boldsymbol{w}_y}} \tag{6.8}$$

The maximum of ρ with respect to $\boldsymbol{w}_x$ and $\boldsymbol{w}_y$ is the maximum canonical correlation.

To calculate the CCA between motion and speech we divided the features into speech and motion features, resulting in a speech vector consisting of the first 12 MFCC coefficients, pitch, energy, and their respective first and second derivatives and a motion vector consisting of the Euler Angles. To compare our analysis with the the analysis done by Busso et al. (2006) we also performed CCA between the Euler angles (3D vector) and energy, pitch and their first and second derivatives (6D vector). The correlations found by our analysis were much lower than the correlations reported by Busso et al. Table 6.4 shows the correlation results between the features. One possible reason for this difference is that Busso et al. included mostly motion features that were located around the mouth in their correlation calculation. This of course gives higher correlation but does not give much insight into the relationship between head motion and speech.

Speech Features	CCA
MFCC, E, F0	0.08
E, F0	0.07

Table 6.4: Frame-wise Canonical Correlation Analysis between Speech and Motion Features

The correlation analysis results indicate that it is not straightforward to model the relationship between speech and head motion as no apparent correlations between the two feature spaces exist. To model the speech and head motion, the temporal properties of the two signals will have be taken into account.

6.5 Speech-based Unit

6.5.1 Phonemes, Syllables, Words, and Phrases

The initial perspective on the problem of speech driven animation was entirely from a speech point of view and therefore a speech-based modelling unit was investigated first. The speech signal changes quite rapidly compared to head motion, which is reflected in the duration of phonemes, as they are quite short. To successfully model head motion, a longer unit is needed. Several possible candidates present themselves

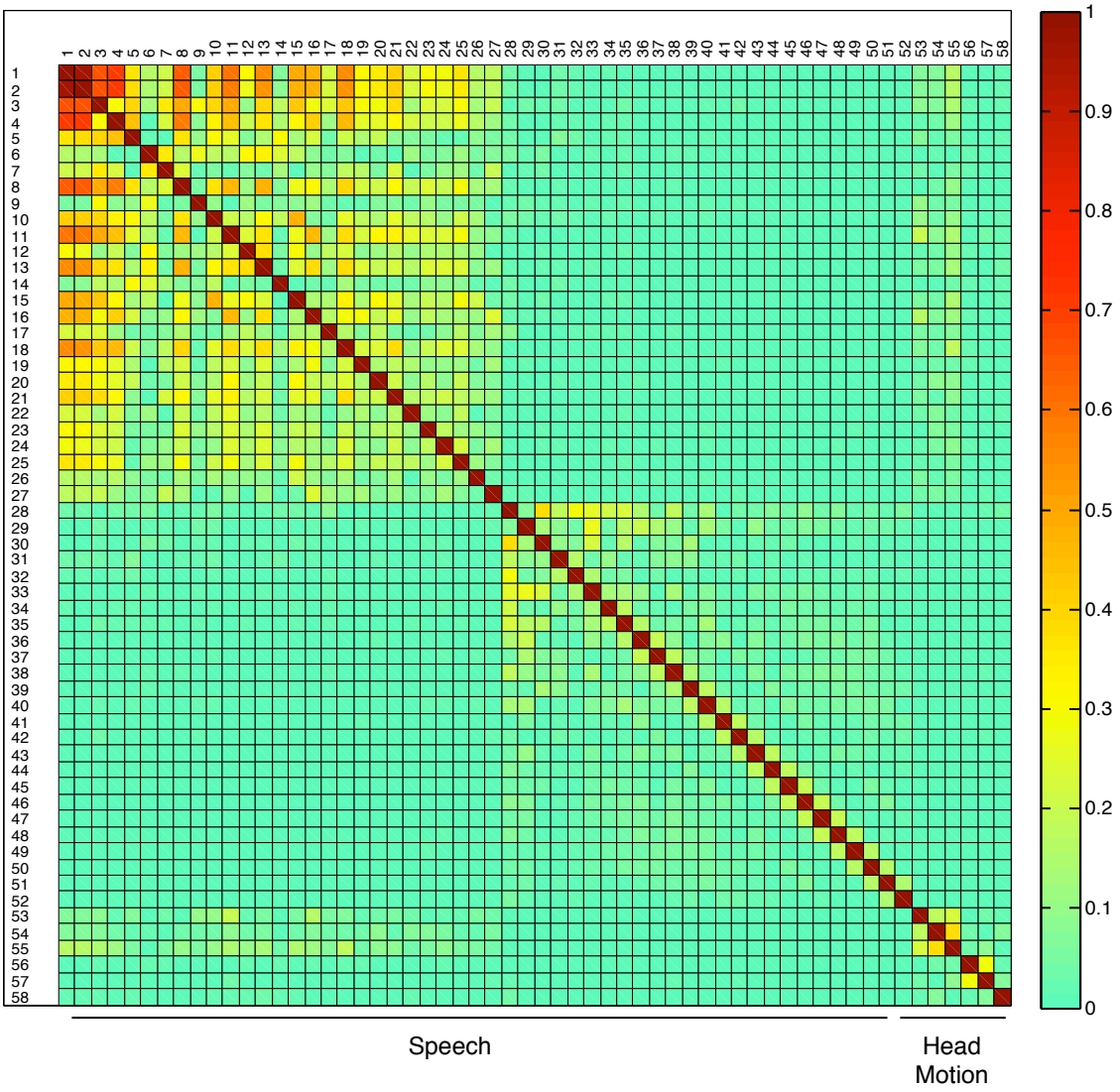

Figure 6.3: Correlations between Euler Angles and speech features. Feature 1-2 are F0 and its first derivative. Feature 3-52 are the first 25 MFCCs and their first derivatives. Feature 53-58 are the 3 Euler angles of head motion and their first derivatives. No correlations between head motion and speech features exist.

in an utterance, starting from phonemes, to syllables, words, phrases, or the whole utterance. While words might be a good modelling unit, it is not straightforward to group them into concise clusters that would make up a unit. Phrases seems like the next best candidate, as they are longer than phonemes and easier to group than words by position in the utterance. Furthermore, phrase breaks constitute natural boundaries in our language, be it in our speech or in other communication channels. Therefore it is reasonable to assume that breaks in the speech signal correlate with breaks in the non-verbal channels, specifically head motion.

6.5.2 Preliminary Data

Because of the lack of other data, the initial models were trained on data that collected in Japan by Keisuke Uematsu. The data were collected with the MotionStar (Ascension Technology Corporation) motion capture system. The measuring method is a magnetic field with a sampling frequency of 86.1 Hz. Two male Japanese speakers were recorded speaking utterances from the ATR corpus that they had previously read. To simulate a real interaction the investigator was seated opposite of the participant who was told to address him. The two persons were about two meters apart. The data was normalised and Euler angles for the head orientation were calculated from the raw sensor data. In addition to the Euler angles, the velocity, also called the delta coefficient, was calculated for each dimension. The delta is used to give a better representation of the movement over time. The data were also fully annotated for phrase breaks that were used to cluster the data. The annotations were done in Japan by staff at ATR.

6.5.3 Modelling Head Motion using Phrases

6.5.3.1 Phrase types

To model the different variations of head motion successfully, the data were divided up into different training and test sets consisting of 600 and 20 utterances respectively. Because the modelling unit is phrases, each utterance was segmented into different phrase types. It is not entirely clear what these phrase types could be, as many alternatives exist on how to group phrases. For the purpose of this study, three different

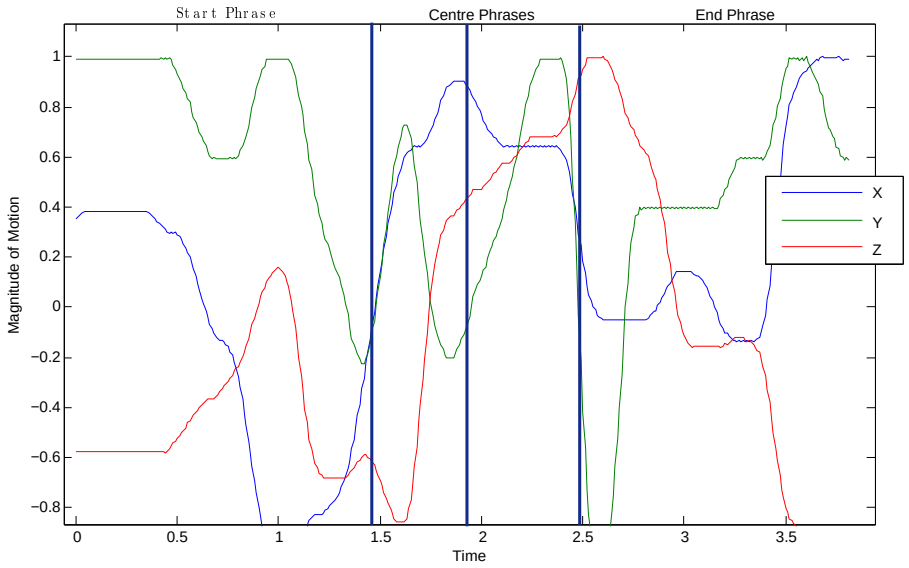

Figure 6.4: An example of head motion in all three dimensions. Lexical phrase boundaries are marked with a vertical bar. The utterance is segmented into a start phrase, two centre phrases, and an end phrase.

types were defined according to their position in an utterance, corresponding to the start phrase, the centre phrases, and the end phrase. Figure 6.4 shows the head motion trajectories of a typical utterance with the phrase breaks marked. The reason why this grouping seems to be valid is that the gesture activity changes in the course of an utterance. The behaviour exhibited at the beginning of utterances is more similar than the behaviour at the beginning and at the end. Previous research in related fields has found that gestures seem to be more concentrated at the beginning of an utterance than towards the end (Dittmann & Llewellyn 1969). It is relatively safe to assume that the same holds true for head motion although as can be seen in Figure 6.4 the phrase breaks and the points of change in the motion to not align. Still, using phrases as a way of segmenting the signal is a good starting point.

6.5.3.2 Training

For training our model the data were segmented into phrase units. A phrase in this case is seen as a unit of gesture that also coincides with a syntactic phrase. Its position in an utterance carries alignment information. We ended up with training examples for three different phrase positions in an utterance: start, centre, and end. Figure 6.4 shows

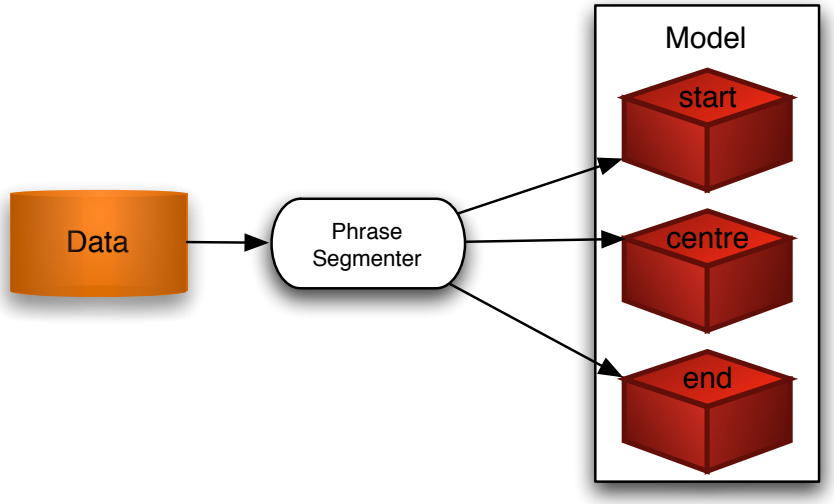

Figure 6.5: The data were segmented into phrase types. One HMM is trained for each type, 'start', 'centre', and 'end'.

how a particular utterance would be divided into different training examples. For each speaker, three HMMs were trained on the Euler angles and their respective deltas from the motion capture data. The HMMs correspond to start, centre, and end phrases. Each HMM consisted of 10 states with two mixture components per state. Training was performed using the conventional EM training algorithm so as to maximise the probability that the model generated the training data. Figure 6.5 illustrates the training process.

6.5.3.3 Synthesis

The modelling units are determined from the speech signal using the previously defined maximisation

$$\boldsymbol{u}^* = \underset{\boldsymbol{u}}{\operatorname{argmax}}\, p(\boldsymbol{O}^S|\boldsymbol{u})P(\boldsymbol{u}) \tag{6.9}$$

In the current case $\boldsymbol{u}$ are syntactic phrases. So the unit sequence $\boldsymbol{u} = (\boldsymbol{u}_1, \ldots, \boldsymbol{u}_N)$ is predetermined by the phrases in the utterance. This unit sequence is then used in the maximisation

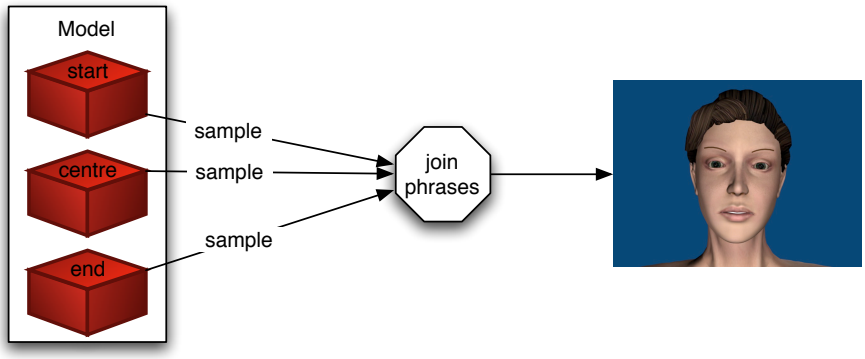

Figure 6.6: During synthesis each model generates a pre determined amount of samples, that are concatenated. The resulting trajectory is used to drive the animation.

$$\hat{\boldsymbol{O}}^M = \underset{\boldsymbol{O}^M}{\operatorname{argmax}}\, p(\boldsymbol{O}^M|\boldsymbol{u}^*) \tag{6.10}$$

to generate the motion observations given the phrase units. Although the ML parameter generation algorithm was used in later experiments, at the time of working on the phrase-based synthesis, only the original version of HTS was available, which was not ready to be used in motion generation. Therefore we employed a smooth sampling method to generate motion trajectories from the HMMs, which had the added benefit of being stochastic, in the sense that no two trajectories generated from the same models were alike.

6.5.3.4 Smooth Sampling

Figure 6.6 gives a basic overview of the synthesis procedure. It consists of several parts that will be described in detail in this section:

1. sample from each phrase model
2. concatenate sample sequences
3. filter samples

The following algorithm generates observations from a stochastic model like an HMM

1. Enter first state of the HMM

2. Select a mixture component randomly to allow for stochastic trajectories
3. Generate a sample based on the probability distribution of the selected component
4. Decide to stay in the state or move to the next state based on the transition probabilities of the current state
5. Repeat 1 to 4 until the last state is exited

There are two major problem with such a basic sampling method: First, The state durations are not specified, meaning that there is no way that the user can specify how many samples the sampler will generate. Second, the drawn samples are independent from each other, which results in non-smooth trajectory.

The former can be solved by making the state durations deterministic. The transition probability determines how many samples a state generates. For example if an utterance is 100 frames long, and our model has 3 states where the first two have a probability of staying in that state of 0.6 and the third of 0.4. We sample from the first and the second 37 times each and from the last state 26 times.

We chose a low-pass filter to filter the output and its filter response can be seen in Figure 6.8 The high frequency components of the signal were filtered out. Figure 6.7 shows an unfiltered signal and a filtered signal and it can be seen that the filtered trajectory is much smoother.

6.5.4 Preliminary Evaluation

6.5.4.1 Initial assessment

Although a full evaluation was performed on the final model in Chapter 7, a short evaluation was done to determine if the proposed method offers an improvement in head motion synthesis over static or random motion and to see which of the goals laid out at the beginning of this chapter were fulfilled. First of all a speech-based unit was used which is human-readable as phrase types are understandable. Then the synthesis method is non-deterministic. Figure 6.9 shows two sequence of frames. They are both for the same utterance over the same time period, but they display different head

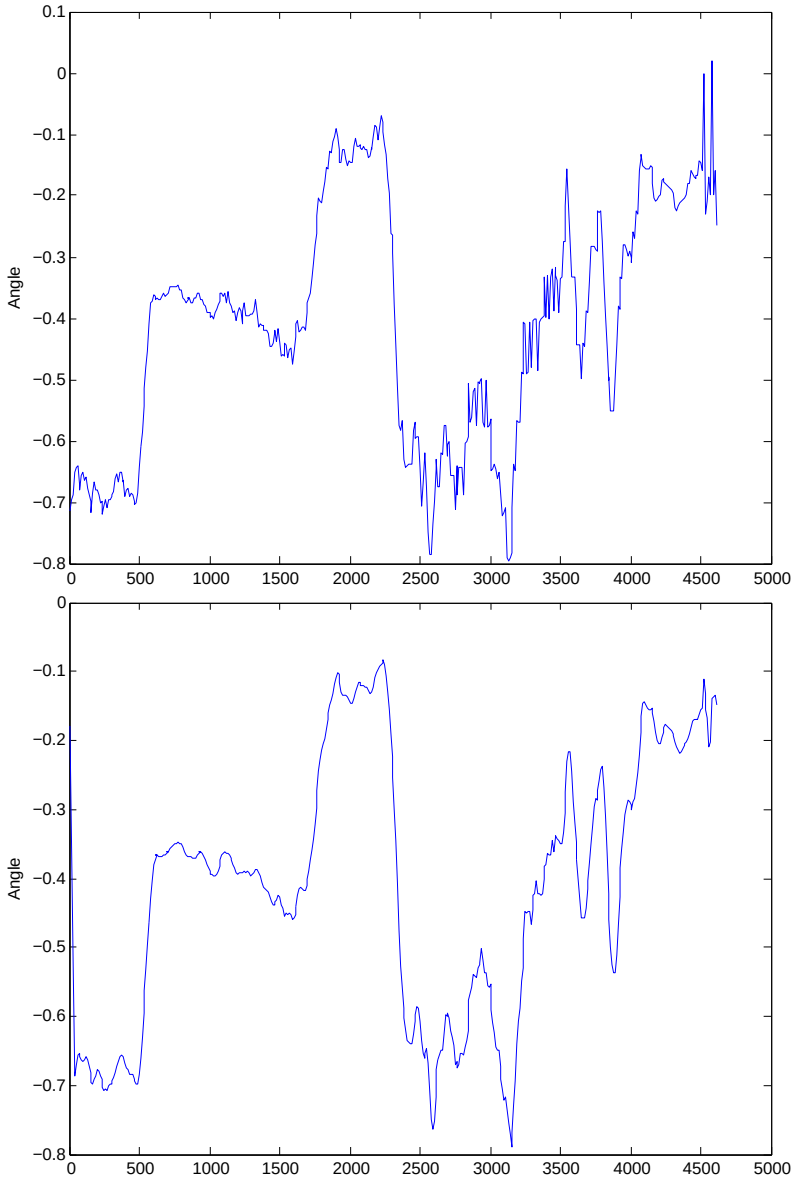

Figure 6.7: Not filtered output on top vs. filtered output on bottom.

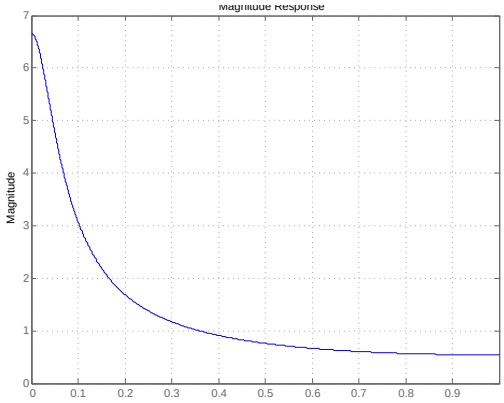

Figure 6.8: Frequency Response of the Filter

Figure 6.9: Sequence of frames from the same utterance but the head motion is different

motion. There is no dynamic range control and the trajectories are discontinuous and need to be smoothed using post-processing.

6.5.4.2 Method

Two different English utterances were animated with speech and randomised saccadic eye motion every. Each utterance was synthesised with static head motions, random head motion, and head motion that was generated by the model. For static head motion, no movement of the head was animated. Random head motion was generated from a Gaussian that was trained on all the training data, reflecting the distribution of the data. The model condition was animated using the stochastic phrase-based generation method outlined above. Five participants were presented with pairings of animations for each utterance. They were asked verbally to identify the animation that they preferred. We did not specifically ask for which one they found more natural but were just interested in their preference. There were a total of three configurations for each utterance and the order was randomised.

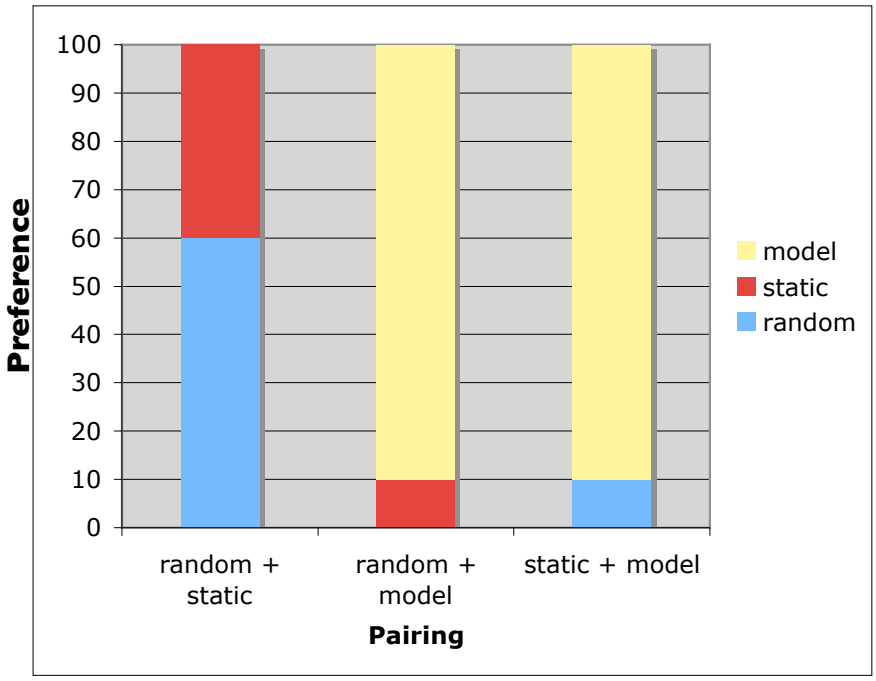

Figure 6.10: Chart shows the results of the pair-wise evaluation for 5 subjects. Each participant were shown 2 animations in succession and asked which one they preferred. The bars shows the number of preferences over the paired condition.

6.5.4.3 Results and Discussion

Figure 6.10 gives an overview of the results. In general movement was preferred over no movement. The model generated movement was preferred over the other styles 85% of the time.

There seems to be a preference for the animation generated by our model over static or random animation, although a random or static baseline would always be beaten by something more sensible. The movement was more interesting than no movement and random head gestures but generally the feedback from the participants suggested that the movement did not look very good. The generated movement, although not specifically evaluated, seemed to have little synchrony between the movement and the speech. We know from observation that speech pauses and movement pauses usually coincide, but the current model, since its phrase based, produces movement during pauses, which leads to the perception of asynchrony. There seems to be more than one type of head motion per phrase, as can be seen in Figure 6.4, where for each phrase several peaks and valleys in the Euler angle trajectories can be identified. Another

problem is that using a phrase-based unit, or any speech-based unit for that matter, does not guarantee that the unit boundaries coincide with movement boundaries because any speech based unit describes only speech. Phonemes describe small building blocks of speech, and phrases describe parts of a sentence but they do not necessarily describe motion boundaries. Therefore in the next section motion-based units are considered as they describe the motion rather than the speech.

6.5.5 Speech-Gesture synchrony

Noticing the absence of synchrony between speech and motion in the motion generated by the previous model, the other alternative, of employing a motion based unit becomes very attractive. One of the reasons for the lack of synchronisation, is that gestures seem to precede speech but not necessarily by a fixed offset. By how the gestures precede the speech is still open to debate. In a review of the literature, Rimé & Schiaratura (1991) found support for the hypothesis that speech-accompanying gestures help to launch the cognitive processes underlying speech. This alludes to a perception point of view where synchrony between speech and gestures would be perceived only if gestures precede the speech. By using speech-based units, this would never be the case as the gesture would have to start before the speech unit boundary. Therefore using a motion based unit, that starts and ends at boundaries of motion, might improve the perceived synchrony.

6.6 Motion-based Units

6.6.1 Optimal Motion Unit

The link of head motion to the speech production process suggests that in order to drive head motion with speech data the temporal relationship between the two streams has to be taken into account. Since frame-wise analysis of the data streams is not sufficient to model temporal relationships, the data have to be segmented into longer parts. Head motion is modelled by introducing a conceptual unit of motion that is based on manual labels that span over several frames. Since the link between the two streams of speech and head motion is not straightforward, a modelling layer is introduced where speech

and motion features are used together to train models that represent units of motion. This approach is described in detail in Chapter 5. It is hoped that the temporal relationship and some of the long range dependencies between the two streams can be captured by this approach. This approach is based on HMMs that act as a sequence generator and can be partially evaluated by an accuracy measure similar to word error rate used in speech recognition. The recognition accuracy provides intermediate results that can be used to determine the optimal unit. The unit can either be based on manual labels or automatically determined by clustering.

6.6.2 Evaluation method

Although evaluating the full system objectively would be the ideal case, with synthesis this is not always possible, as often a gold standard is missing for the output. In the case of head motion, there are many possible natural trajectories for a given utterance, and comparing a synthesised one to a recorded one might not provide a lot of insight into its naturalness. Furthermore, it is not straightforward to compare trajectories. Root mean squared error or methods that compare the alignment of certain events in the original with the generated trajectory are not reliable, as the correlation between motion and speech is far from frame wise. It is possible on the other hand to partially evaluate the proposed method objectively by employing an accuracy measure of the predictive part of the system. The accuracy is calculated in a similar manner to word error rate in speech recognition:

$$Acc[\%] = \frac{\#of correct labels - \#of insertions}{Total\ number\ of\ labels} \times 100$$

All the presented recognition experiments were conducted using the free speech data from Speaker 1. In total the data were divided into seven free speech segments which were approximately three minutes long each, giving a total of about 20 minutes. In the following recognition experiments the models were trained on six segments and recognition was performed on seventh held out segment. To be able to fully evaluate the models, cross validation was performed where every set was held out once and the models were trained on the remaining sets. The presented results were an average of the seven experiments.

In addition the generated motion will have to be evaluated perceptually. Chapter 7 describes that evaluation.

6.6.3 Manual labels

To be able to better model the relationship between speech and head motion, the collected data was manually labelled to describe segments of distinct motion. Labelling head motions is not straightforward as for example gait, where motions can be labelled as running, walking, dancing, etc. Head motion does not seem to have any clear distinctions between different movements, therefore we decided to use the most basic motions that can be seen in the data. The following four basic labels were invented:

1. shift: the head shifts axis of movement
2. shake and nod: lateral movement around one axis
3. pause: no movement / rest position
4. default: slow movement / small dynamic range

In particular the video recordings and the actual movement trajectories were inspected and the following guidelines were used to label the data:

- **shifts** are movements that start at one place and end at another, with periods of no or very little movement before and after the shift. For example the head rests straight on and then tilts slightly to the side, staying in that position for over a second, would constitute a shift.
- **shakes** are movements that start and end in one place. An example of a head shake is where the head does not come to rest in between the left and right motion.
- **pauses** are periods in the data, where there is absolutely no movement at all. An example is a person looking straight on into the camera without moving.
- **default** movements are sections in the data, where there is a small dynamic range in the movement. This type of movement is the natural human motion that is related to breathing or other activities.

Figure 6.13 shows typical shake and shift motions in the Euler angle representation. If the movement was not distinct a default label was applied. Figure 6.11 shows the distribution of labels in the data and Figure 6.12 their average length for speaker 1. From the figures it is clear that the default label is longer than the other labels. What

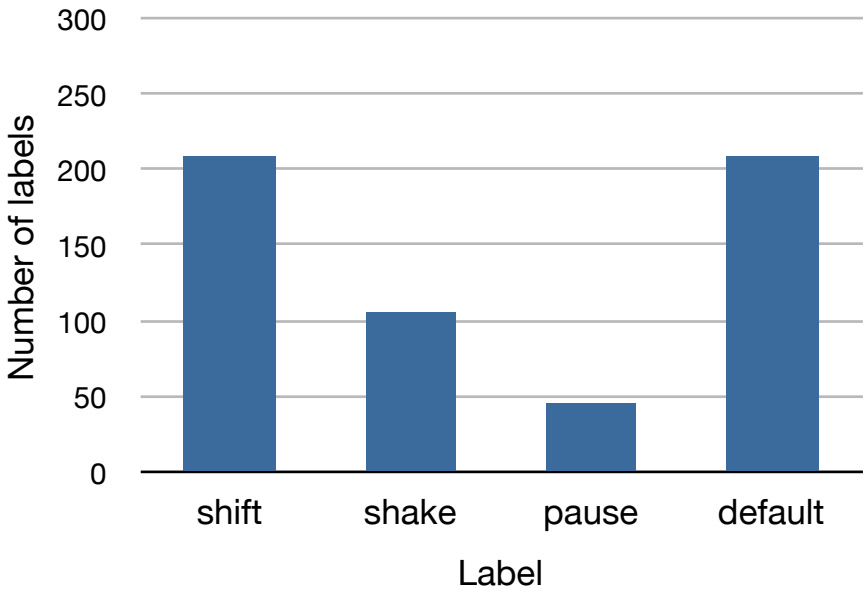

Figure 6.11: Distribution of the number of different labels in the 20 minute free speech for speaker 1.

is interesting is that it is not much more common than the other labels, confirming that the selected speaker had varied head motion.

6.6.3.1 Distinct Head Motion Labels

It is clear that the newly defined labels need to be different from other types of labels (e.g. phonemes) in the sense that they need to capture variety in the data that can not be captured with other labels. To test for the distinctiveness of the proposed label set from already existing label set, the average mutual information between phonemes, the viseme set described in Section 6.2.1, and the manual head motion labels was calculated.

Mutual information is a measure of the dependency between two random variables and defined as

$$I(X;Y) = \sum_{x \in X, y \in Y} P(x,y) \log \frac{P_{XY}(x,y)}{P_X(x) P_Y(y)} \tag{6.11}$$

where $P_X(x)$ and $P_Y(y)$ are the marginal probability distribution functions of X and Y

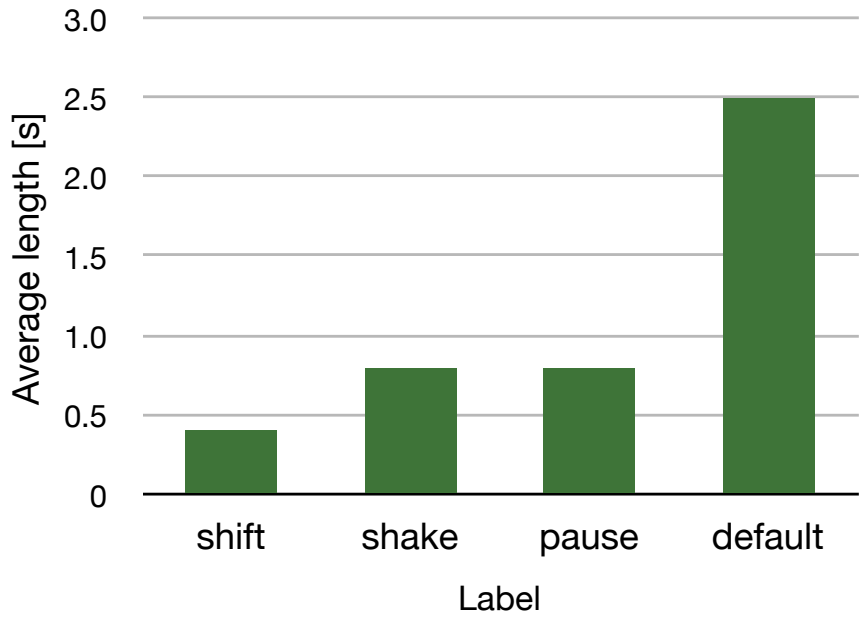

Figure 6.12: Average length of each label for 20 minutes of free speech for speaker 1.

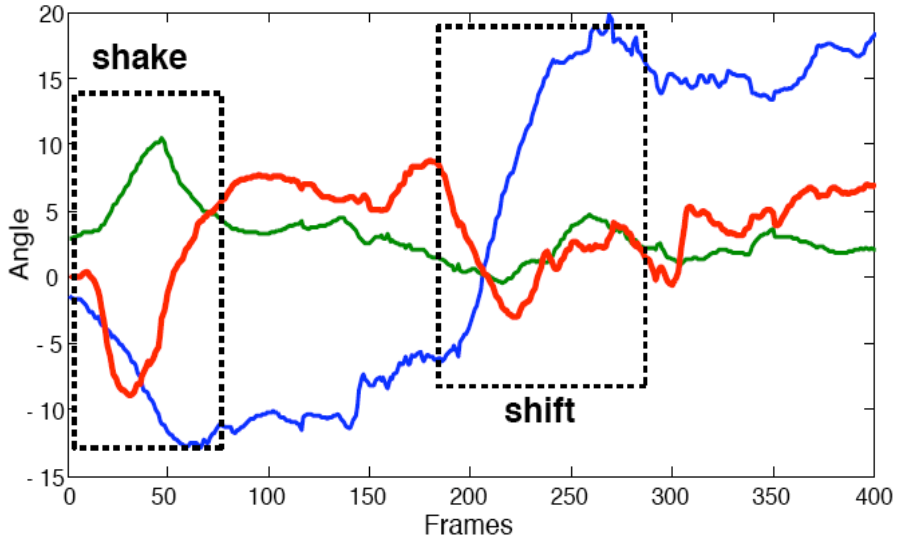

Figure 6.13: Example of a shift and shake as indicated by the marked region.

respectively. Mutual information is defined in terms of entropy $H(X)$. e.g.

$$H(X) = -\sum_{x \in X} P(x) \log P(x) \tag{6.12}$$

where $I(X;Y)$ can also be expressed as follows:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) && (6.13)\\ &= H(Y) - H(Y|X) && (6.14)\\ &= H(X) + H(Y) - H(X,Y) && (6.15)\\ &= H(X,Y) - H(X|Y) - H(X|Y) && (6.16) \end{aligned}$$

where $H(X|Y)$ is the conditional entropy. The conditional entropy is the entropy of a random variable X given the random variable Y. It is defined as

$$\begin{aligned} H(X|Y) &= -\sum_{y \in Y} P(y) \sum_{x \in X} P(x|y) \log P(X|Y) && (6.17)\\ &= -\sum_{y \in Y} \sum_{x \in X} P(x,y) \log P(x|y) && (6.18)\\ &= -\sum_{y \in Y, x \in X} P(x,y) \log P(x|y) && (6.19) \end{aligned}$$

The results in Table 6.5 were calculated using the mutual information as defined above. The mutual information was calculated using 20 minutes of the read speech data that was annotated for head motion described in Section 3.4. It shows that the defined head motion labels have very little resemblance to either visemes or phonemes. This is further evidence that head motion is better modelled with motion based units than with speech based units.

Furthermore, the head motion labels were analysed by calculating the mutual information between the proposed head motion labels and the phrase based labels described in Section 6.5.3.1. The results are shown in Table 6.6. The analysis was done for only the data where phrase transcription was available, in this case 15 utterances of read data. The distribution of data is shown in Table 6.7.

Although different data sets were used to calculate the results in Table 6.5 and Table 6.6 it is interesting that the mutual information between head motion labels and phrases seems to be higher than between head motion labels and phonemes. One reason for

X	Y	
(head motion)	phoneme	viseme
$H(X)$	1.84	1.84
$H(X\|Y)$	1.60	1.61
$I(X;Y)$	0.24	0.24
$I(X;Y)/H(X)$	0.13	0.13

Table 6.5: Mutual information between phonemes, visemes, and manual head motion labels. The mutual information was normalised by $H(X)$ e.g. $I(X;Y)/H(X)$, where X are the head motion labels, and Y are either phonemes or visemes.

$H(X)$	$H(Y)$	$H(X\|Y)$	$I(X;Y)$	$I(X;Y)/H(X)$
1.56	1.66	1.26	0.30	0.19

Table 6.6: Mutual information between phrase types and head motion labels. The mutual information was normalised like $I(X;Y)/H(X)$, where X is the head motion labels, and Y is the phrase types.

	shift	shake	pause	default	start phrase	centre phrase	end phrase
no of samples	27	18	14	61	15	22	15

Table 6.7: Distribution of head motion labels and phrase types in the data set used for the mutual information calculation.

this could be that timing in the utterance is important. Since the phrases were divided by location in the utterance, it might be possible to predict the head motion a bit better than with phonemes, where no timing information is given. Still, the results indicate that using phrases as a unit is not adequate as there is little evidence that it is possible to predict different types of head motion from them. The manual labels, on the other hand, are guaranteed to label distinct motions differently.

6.6.3.2 System Overview: Modelling Speech and Motion simultaneously

The modelling approach is based on the notion that head motion can be divided into a number of short homogeneous units that can each be modelled individually. For example all the data labelled as shift are modelled by one model and the data labelled as shake are modelled by another model. Statistical models are trained for each label that are used to predict a unit sequence from speech data. For each input sequence of speech frames, a sequence of motion labels is produced that are chosen by the most likely sequence of models. Figure 6.15 shows an example of an utterance and the corresponding generated label sequence. An overview of the proposed system can be seen in Figure 6.14.

During recognition only the speech features were used to determine the sequence of motion labels. Employing a framework that is similar to speech recognition, using the accuracy measure defined in Section 6.6.2 allowed us to evaluate different aspects of the modelling process in a more principled way.

6.6.3.3 Model selection

Several possible models were proposed in Chapter 5 and two specific models were evaluated in this section. The models shown in Figure 5.5 and Figure 5.6 respectively are different in respect to synchrony. The former models speech and motion unit synchronously where as the latter models it state synchronously. In practice this means speech and motion are being modelled with different HMMs or within the same HMM. A pilot experiment compared the recognition accuracies of a model trained on both motion and speech and a model trained on speech. It was found that the model trained on both streams performed better (Acc=67) than the model trained only on speech (Acc=63). One reason for this increase in recognition accuracy could be hat by

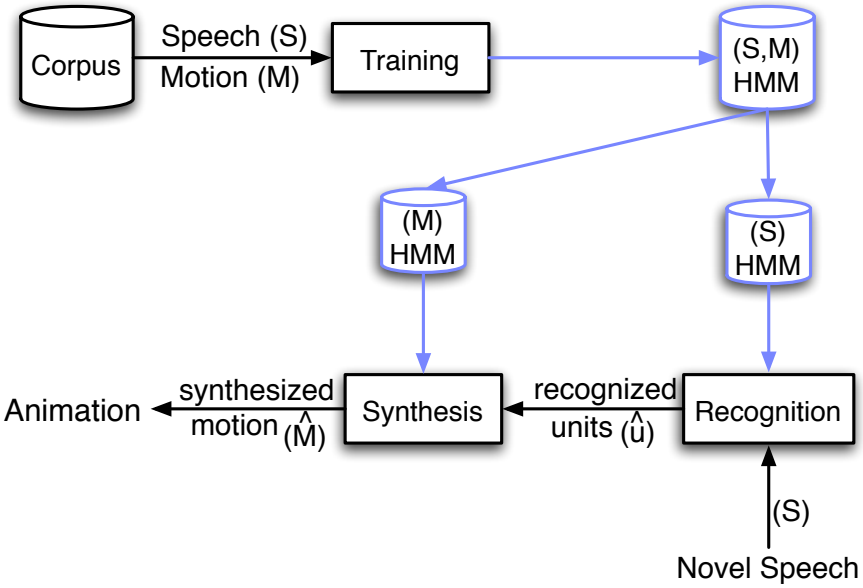

Figure 6.14: System Overview: The system is trained on parallel speech and motion data that has been labelled for head motion, resulting in models for each motion unit. To synthesise, novel speech is used to estimate a sequence of motion labels. The parameter generation algorithm produces motion trajectories from the model sequence that corresponds to the estimated motion units sequence.

training on both streams, the transition probabilities can take both streams implicitly into account during recognition, which improved results. One reason for this could be that since we are recognising motion units, the length of the motion units is better represented when trained on both feature types.

6.6.3.4 Model parameters

We conducted experiments with the models based on the manual labels to find the optimal model parameters. The actual test results were obtained by 7 fold cross validation using a split of approximately 18 minutes training and 2 minutes testing data. The results shown are the average results of this cross validation.

Motion and speech change at different rates, where speech changes faster than motion. The motion labels are longer than phonemes to reflect this difference in rate of change. The motion labels are between 0.8 and 2.5 seconds long, compared to phoneme units which can be between 0.025 seconds and 0.150 seconds long. To determine the op-

Figure 6.15: This figure shows predicted and actual labels for the utterance: "He died in 1996 I think it was, um, my grandmother still has his um...". From the given information, FO and energy of the speech (green) a unit sequence is predicted (white). The real unit sequence (grey) and its corresponding head motion Euler angle trajectories (yellow) are shown at the bottom of the figure.

timal model length, the number of states was gradually increased. From the results shown in Figure 6.16 it seems clear that around 16 states per model are required for adequate recognition performance. The optimal number of mixture components was also evaluated and the results are shown in Figure 6.17. Around 8 mixture components yield the best results. Although this seems like a high number of mixture components, the variance of the data described by each label is quite large, as only 4 labels are used. Therefore employing a large number of mixture components per state helps to model the variance found in the data.

Furthermore, since the default label has somewhat a special status as a class that incorporates everything that could not be described the other classes, the number of mixture components in the default model was evaluated independently. The variance of data labelled as 'default' is higher than for the other labelled data. By increasing the number of mixture components in the default model the variance in the data can be modelled better. Experiments were carried out where the number of mixture components in each model was 4 per state and only the default model's mixture components were increased. The results presented in Figure 6.18 suggest that increasing the number of mixture components in the default model above 8 mixture components does not improve the overall accuracy. It is interesting that by only increasing the number of default model mixture components, about the same accuracy can be achieved as when the number of mixture components of all models are increased. The reason for this could be that the variety of the default label is much higher and therefore only the discriminatory power of that model has to be increased.

6.6.3.5 F0 Feature

The final model topology had 18 states per model and 4 mixture components per state in all models except the default model which had 8 mixture components. In addition the influence of F0 on the model was tested, as F0 was attributed a great significance in the relationship between head motion and speech Yehia et al. (2002). Table 6.8 shows a comparison between models on a feature set that included F0 and models that were trained on a feature set without F0. Finally a model that just used F0, energy and their first and second derivative was constructed as well. The results are also shown in Table 6.8.

		MFCC + E	MFCC + E + F0	E + F0
4 class	Accuracy	67.99%	68.68%	50.38%
	Standard Deviation	2.97	6.32	11.88
	Maximum Accuracy	71.43%	75.41%	69.54%
	Minimum Accuracy	63.95%	61.05%	39.17%
2 class	Accuracy	74.19%	75.54%	73.06%
	Standard Deviation	2.71	4.34	3.24
	Maximum Accuracy	77.16	80.33	76.98
	Minimum Accuracy	69.77	70.00	68.81

Table 6.8: Seven-fold cross validation results for models trained on different speech feature sets. All the data came from speaker 1. The table shows the maximum an minimum accuracy achieved over all folds. Results are presented 2 classes and 4 classes. The first and second derivative of each feature was also used. It is interesting to see that although the results of the 2 class and 4 class tests are not directly comparable, the variance in the results for E+F0 is much higher for the 4 classes than for the 2 classes. This suggests that F0 works well for some utterances but not for all.

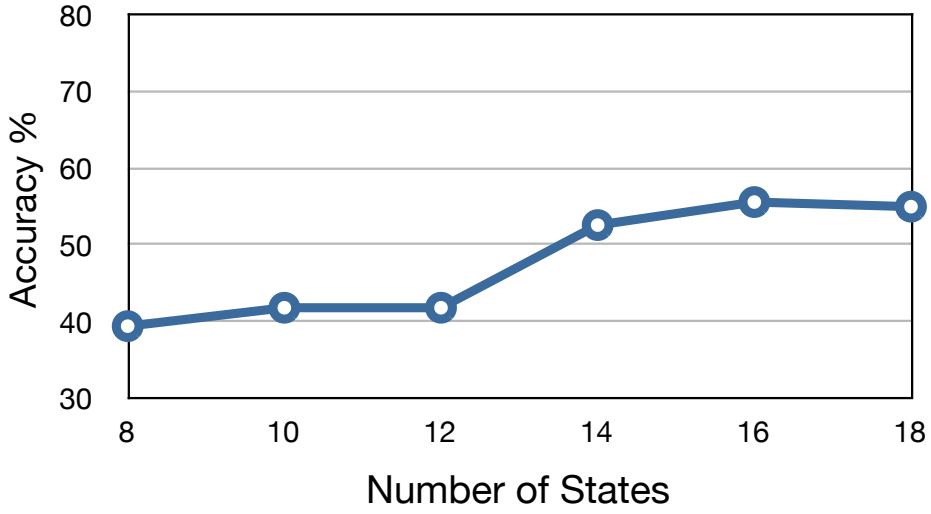

Figure 6.16: Results for different number of states per model. The number of mixture components in all models was 4 except for default where the number of mixture components was 8. The optimal number of states in terms of accuracy seem to be 16. The models were trained on data from speaker 1 and evaluated using 7 fold cross evaluation.

The discriminatory power of the model between regions of high activity (shake and shift) and low activity (default and pause) was tested as well, termed 2 class. The results are shown in Table 6.8 and suggest that the model can distinguish reasonably well between default/pause segments and other regions.

6.6.3.6 Unit variation

Although it is difficult to increase the number of basic labels as it is not straightforward to describe Euler angle configurations, a number of methods were tried to split the basic labels into more categories. First the labels were grouped according to time, meaning that different versions of the same basic labels were constructed based on their duration. The threshold at which the point the labels were split was chosen to be the median of the duration of each label. Another method of incrementing the number of labels, grouped the basic labels along the dimension which had the highest dynamic range. For example if the 'yaw' dimension exhibited the largest dynamic range for that particular 'shift' segment, it was given the 'yaw shift' label. If on the other hand an-

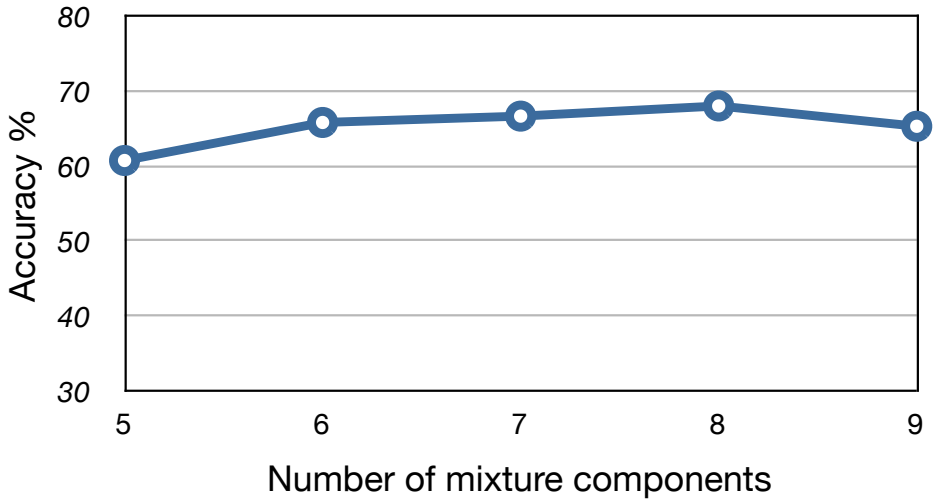

Figure 6.17: Results for different number of mixture components. The number of states in all models was 16. 8 mixture components seem to yield to the best results. The models were trained on data from speaker 1 and evaluated using 7 fold cross evaluation.

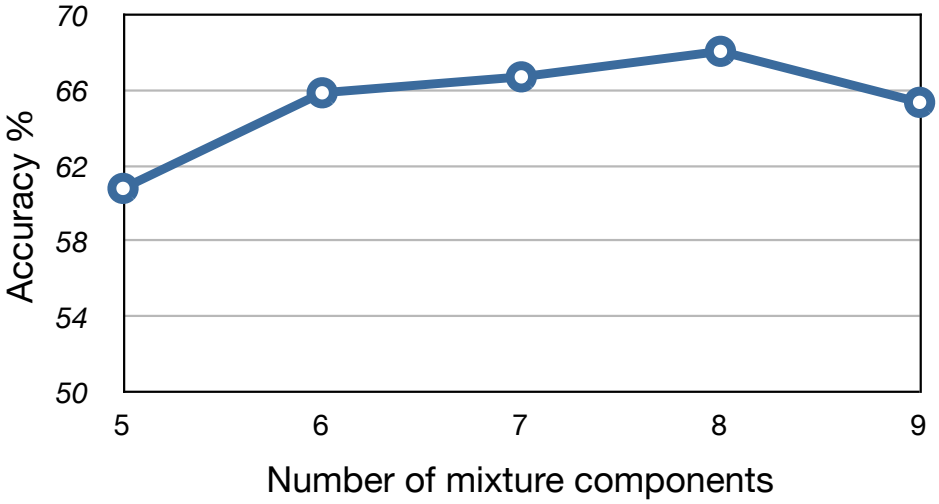

Figure 6.18: Results for different number of mixture components for the default model. The number of states in all models was 16. The models were trained on data from speaker 1 and evaluated using 7 fold cross evaluation.

	grouped according to the duration of each unit	grouped according to dynamic range of each dimension	standard units
Accuracy for 4 labels (extended labels)	56.89 (39.52)	58.08 (42.51)	69
Number of models	6 + pause	9 + pause	3 + pause

Table 6.9: Recognition results for different number of labels. The results were calculated on the extended label set described in Section 6.6.3.6 and for the standard label set. To be able to compare the results the predicted extended labels were mapped back to the standard label. The prediction accuracy for the extended sets are shown in parentheses.

other 'shift' segment had the largest dynamic range in the 'roll' dimension, it was given the 'roll shift' label. A number of experiments were carried out to test if the increased label numbers helped to capture more variety in the data. All the labels, except 'pause' were split as that label has almost zero change in each dimension. The accuracy was calculated for both the increased number of labels and the standard number of labels. For example the labels: 'yaw shift', 'roll shift', and 'pitch shift' were treated as just a 'shift' label in the second condition. Recognition results were compared by mapping the extended label sets back to the original ones. The specific recognition results shown in Table 6.9 suggest that duration and dimension of change are not factors that contribute much to the variety in the data as the recognition accuracy drops compared to the standard models. Other factors like context might be more important.

6.6.3.7 Context-dependent manual labels

Finally context-dependent models were trained with one left and one right context. Tree clustering of states was performed to minimise data sparsity problems, which although only four labels were used, were present.

Figure 6.19 shows the recognition accuracy of the context dependent models in comparison to the context-independent models. Overall context dependent models achieve a higher recognition accuracy, which is to be expected as more variety in the data is captured. What is interesting is that, although context improved overall performance, there is still room for improvement.

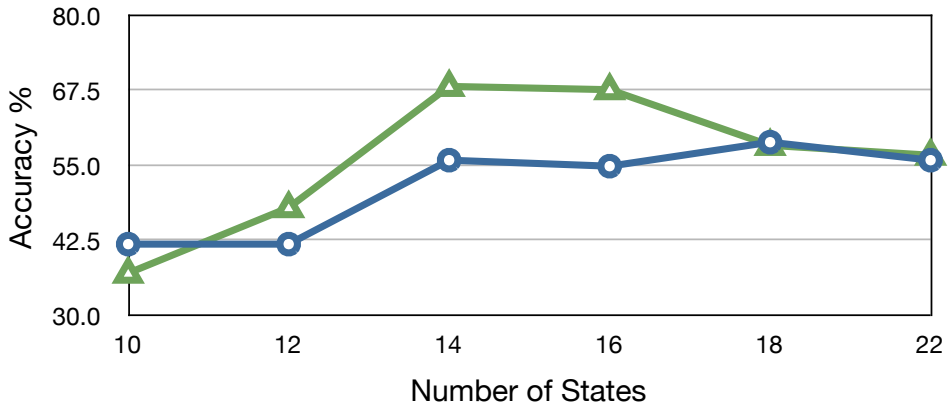

Figure 6.19: Recognition Accuracy for context-independent (CI) and context-dependent (CD) models in relationship to the number of states.

6.6.4 Automatically determined Labels

To build a baseline model, LBG clustering was used to divide the 3D space of Euler angles into K clusters. This was done frame-wise at a frame rate of 500Hz. To compare the results with the manual labels, $K = 4$ clusters were used. If the LBG algorithm clustered more than 5 consecutive frames with the same cluster index, these frames were treated as a sequence. The minimum length of a sequence was therefore 10 ms. Each sequence was labelled with its corresponding cluster index. The cluster indices were treated like labels and the training data, consisting of speech and motion was marked accordingly. One HMM was trained per cluster index on the marked sequences and recognition experiments were performed. The model configuration for each label was 18 states with 4 mixture components per state. This configuration was determined experimentally.

In addition to the LBG clustering, GMM clustering was also performed with K=4 clusters. As can be seen in Figure 6.20 and Figure 6.21 using GMM clustering yields higher recognition accuracy than using LBG clustering. All in all, hand labels still yield better results.

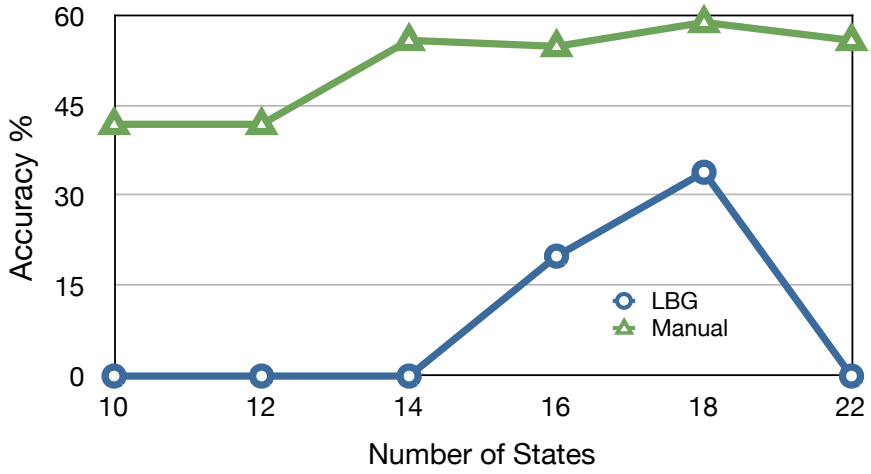

Figure 6.20: Recognition Accuracy for LBG labels and Hand labels in relationship to the number of states.

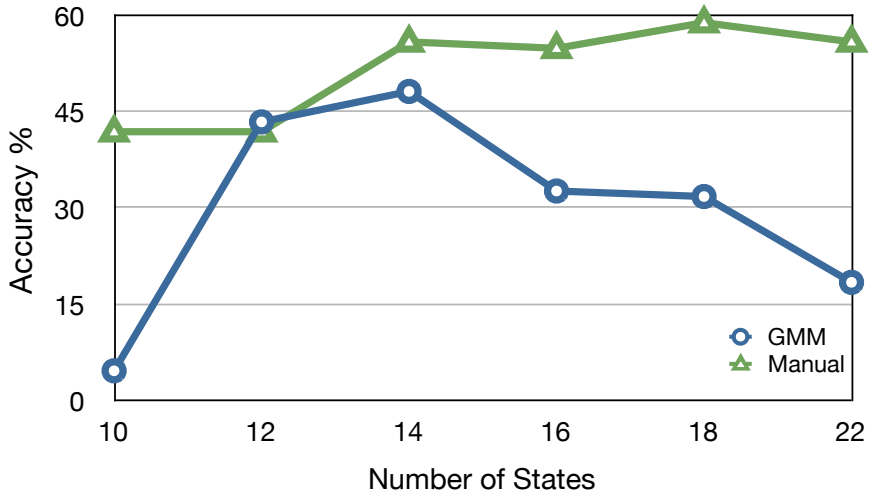

Figure 6.21: Recognition Accuracy for GMM labels and Hand labels in relationship to the number of states.

6.6.5 Summary & Discussion

A number of experiments on finding the optimal head motion unit in terms of recognition accuracy were presented in this section. The models were trained on data manually annotated for head motion and they were able to predict motion labels with accuracies reaching 70%. The optimal models in terms of accuracy had more states than the standard 3 or 5 state speech HMMs. Head motion has a slower rate of change but since speech was modelled synchronously to the motion, the sample rate for both streams was 200Hz. Therefore each unit had up to 200 frames, increasing the number of states needed to model the signal. Adding context and using up to 8 mixture components also helped with recognition, because this could model the variety in the data better. Increasing the number of units was investigated but no gains in accuracy were found. Furthermore it was found that F0 only really helps in distinguishing regions of high activity (e.g. shake, shift) from regions of low activity (e.g. pause, default). When the model was tested for how well it could distinguish regions of high activity and regions of low activity, a model trained only on F0 and energy was able to perform almost on par with models trained on the full feature set. Results hardly improved when the full label set was used. This is in contrast to Yehia et al. (2002), who claimed that they found high per utterance correlation between the Euler angles of head motion and F0. One reason for this discrepancy between previous findings and our findings is that F0 is notoriously difficult to work with because it is a discontinuous signal. In the future a discreet representation of F0 could be investigated, that relates to higher level concepts like information structure in an utterance. Head motion might have more correlation with such concepts than with a low level representation of F0. Another interesting result is that a system based on manual labels was compared to a baseline based on automatically determined labels and outperformed it.

6.7 Head Motion generation

The mapping algorithm defined in Section 5.3 is applied to head motion generation from speech. For head motion it can be described formally as follows. A head motion vector sequence $\boldsymbol{O}^H = (\boldsymbol{o}_1^H, \boldsymbol{o}_2^H, \ldots, \boldsymbol{o}_T^H)$ is generated from a given speech vector sequence $\boldsymbol{O}^S = (\boldsymbol{o}_1^S, \boldsymbol{o}_2^S, \ldots, \boldsymbol{o}_T^S)$ with a length of T frames. By assuming that head

motion units can be predicted accurately, the motion-unit sequence $\boldsymbol{u}^H = (u_1^H, ..., u_{e'}^H)$, which represent the head movements corresponding to the given speech sequence can be incorporated. Using the motion labels units, the first optimisation regarding head motion can be approximated by

$$\hat{\boldsymbol{O}}^H = \underset{\boldsymbol{O}^H}{\operatorname{argmax}}\, p(\boldsymbol{O}^H|\boldsymbol{O}^S) \tag{6.20}$$

$$= \underset{\boldsymbol{O}^H}{\operatorname{argmax}} \sum_{\boldsymbol{u}^H} p(\boldsymbol{O}^H|\boldsymbol{u}^H, \boldsymbol{O}^S) p(\boldsymbol{O}^S|\boldsymbol{u}^H) P(\boldsymbol{u}^H) \tag{6.21}$$

$$\simeq \underset{\boldsymbol{O}^H}{\operatorname{argmax}}\, p(\boldsymbol{O}^H|\hat{\boldsymbol{u}^H}) \tag{6.22}$$

where

$$\hat{\boldsymbol{u}^H} = \underset{\boldsymbol{u}^H}{\operatorname{argmax}}\, p(\boldsymbol{O}^S|\boldsymbol{u}^H) P(\boldsymbol{u}^H) \tag{6.23}$$

Thus we recognise the head motion units $\boldsymbol{u}^H$ from the given speech data $\boldsymbol{O}^S$ using the Viterbi algorithm. For the probability $P(\boldsymbol{u}^H)$, we use back-off bi-gram models estimated from the training database.

Once the optimal unit sequence is determined the parameter generation algorithm described in Section 4.3.2 is applied. An optimal trajectory is generated for the determined unit sequence. Figure 6.22 shows a string of units and the corresponding trajectory.

The resulting trajectories are then used to directly animate the head motion of a talking head. The whole process from determining a model sequence to animation is shown in Figure 6.23. Some example frame sequences from the deterministic and from the stochastic sequence can be seen in Figure 6.24. The head orientation for all three sequences is very similar in the start and end frame where as the orientation of the head in the center frames is different for the three sequences. This is similar to human head motion as we can express the same utterance in many different ways, while preserving some basic idiosyncratic quality.

Furthermore the dynamic range can be controlled using the global variance (GV) constraint on the parameter generation as described in Section 4.3.3. Setting the GV higher will result in a trajectory with higher dynamic range. Finally to generate more varied movements the non-deterministic parameter generation algorithm described in Section 5.4.2.1 is also applied. Both the deterministic parameter generation and the non-deterministic parameter generation are evaluated in Chapter 7.

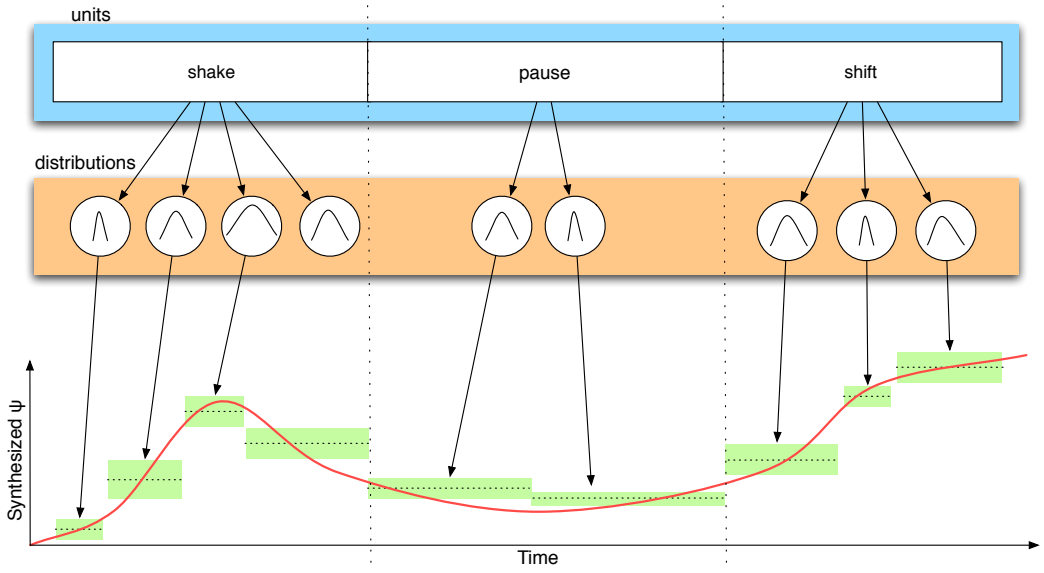

Figure 6.22: The predicted unit sequence produces a distribution sequence, where the parameter generation algorithm converts the distributions into a smooth trajectory. The mean and variance of the distribution influence the shape of the trajectory. The width of the bars represent the variance of each distribution.

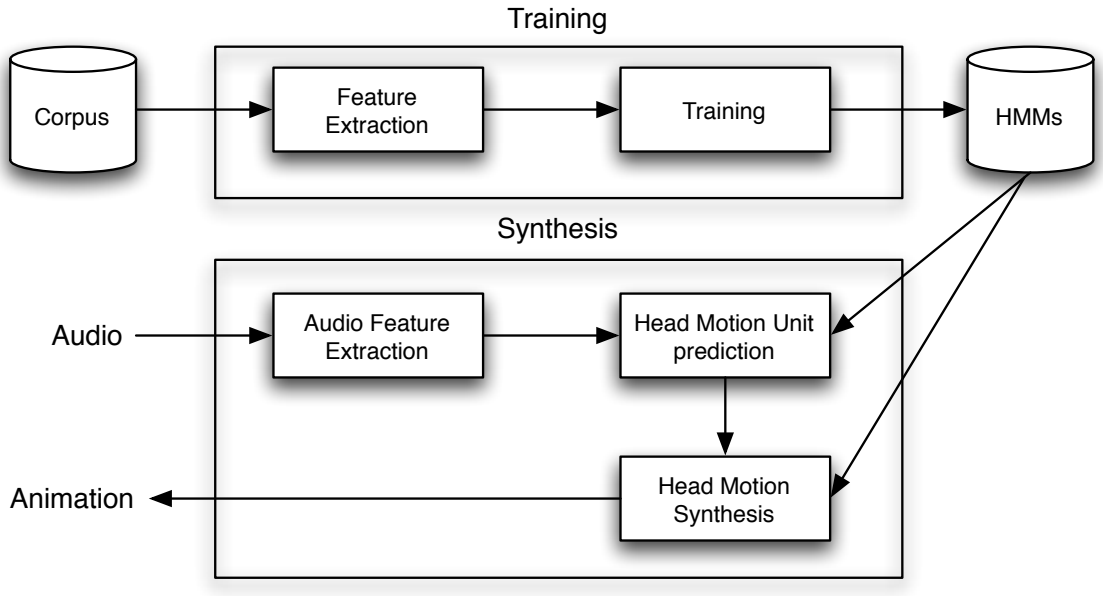

Figure 6.23: Head motion is generated by first determining a head motion unit sequence from the input speech. A motion trajectory is generated from models corresponding to the unit sequence.

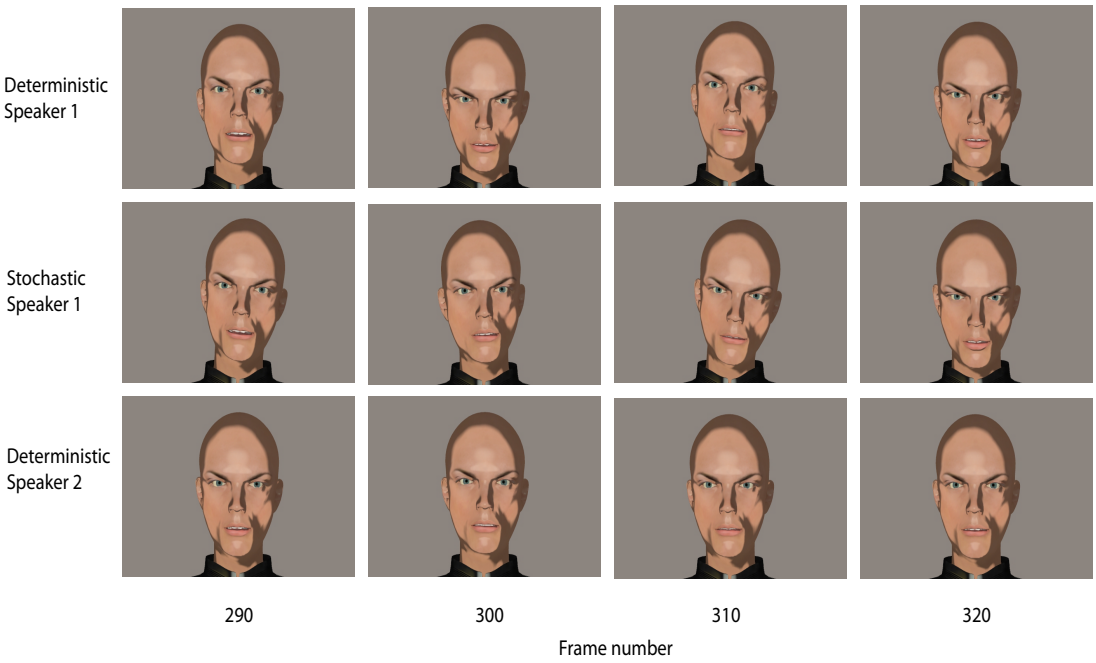

Figure 6.24: Example frame sequences of the different synthesis methods for the utterance: "He was a mountaineer." The top frame sequence shows the output from the deterministic models. The center sequence shows the output of the stochastic model for the same utterance. Notice the difference in head orientation between the top and center sequence. The bottom sequence is the output from a model trained on a different speaker.

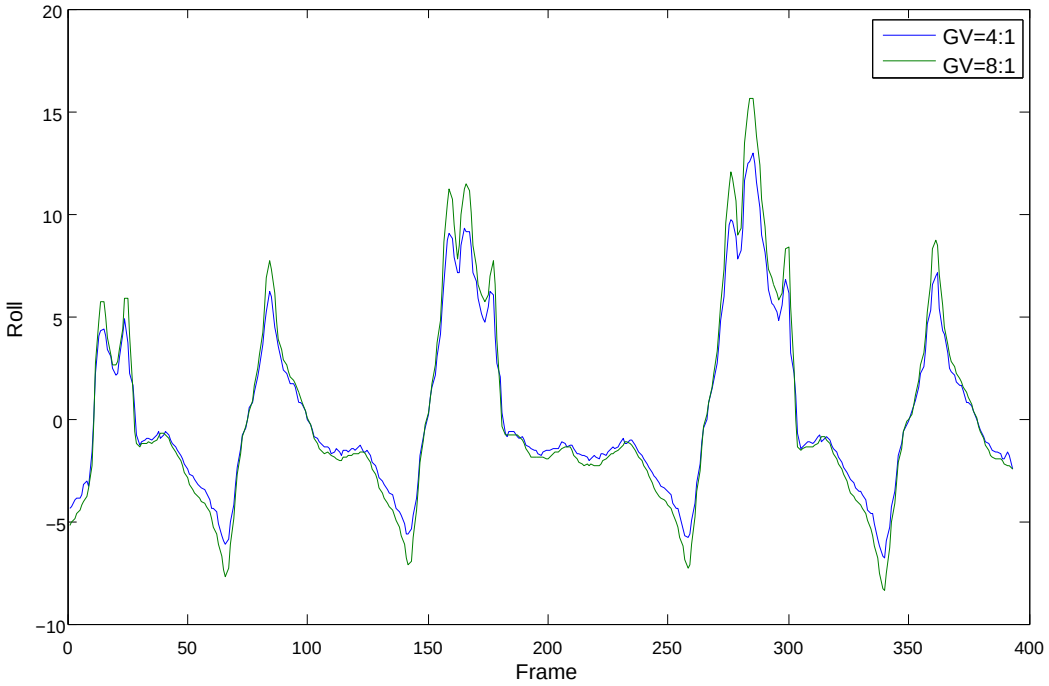

Figure 6.25: The same utterance synthesised with different GV ratio. The ratio is the weighting between the model and the global variance during the parameter generation.

6.7.1 Synthesis Results

The result of the synthesis are presented in this section. The synthesis process has been enhanced from the standard trajectory generation. It is possible to control the dynamic range and to generate trajectories in a non-deterministic way.

First Figure 6.25 shows the synthesised trajectories for different GV ratios. It shows clearly that the two trajectories have peaks and valleys at the same points but the extreme points seem to further apart. Both trajectories are synthesised from the same models, but their dynamic range is controlled by changing the weighting between the model and the global variance.

Non-deterministic trajectory generation is also possible and is shown in Figure 6.26. It shows that both trajectories are different, although synthesised from the same models. The selection of mixture components is done randomly for each state during parameter generation, creating a trajectory that is still "sensible" but non-deterministic.

To show that our models work for different speakers, the same unit sequence is used

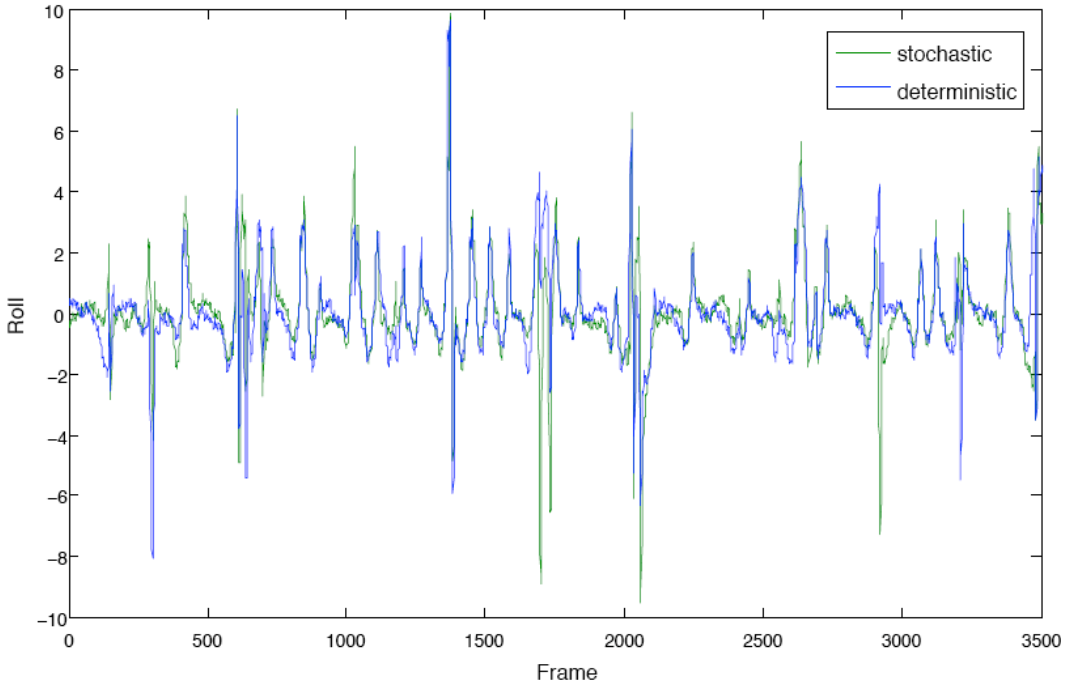

Figure 6.26: Stochastic generation and deterministic generation.

for synthesising from models of two different speakers. The first speaker exhibited a much larger dynamic range and more movement overall, where as the second speaker moved always but with a small dynamic range. These characteristics are also present in the synthesised trajectories as can be seen in Figure 6.27.

6.8 Joint Head Motion and Lip-synchronisation

The lip motion and the head motion synthesis methods described in the previous chapter and this chapter are based on the assumptio that it is possible to determine a motion unit sequence from the speech signal. Combining the lip motion generation and the head motion generation generalises the mapping method and increases the performance of the model. After all adding more information to a recogniser usually helps. An experiment confirmed this. Adding lip motion features to the speech features and using the optimal system as determined by the previous experiments increased the results to 71.2% accuracy, also shown in Table 6.10.

We propose a hierarchical system that is able to generate more than one type of speech

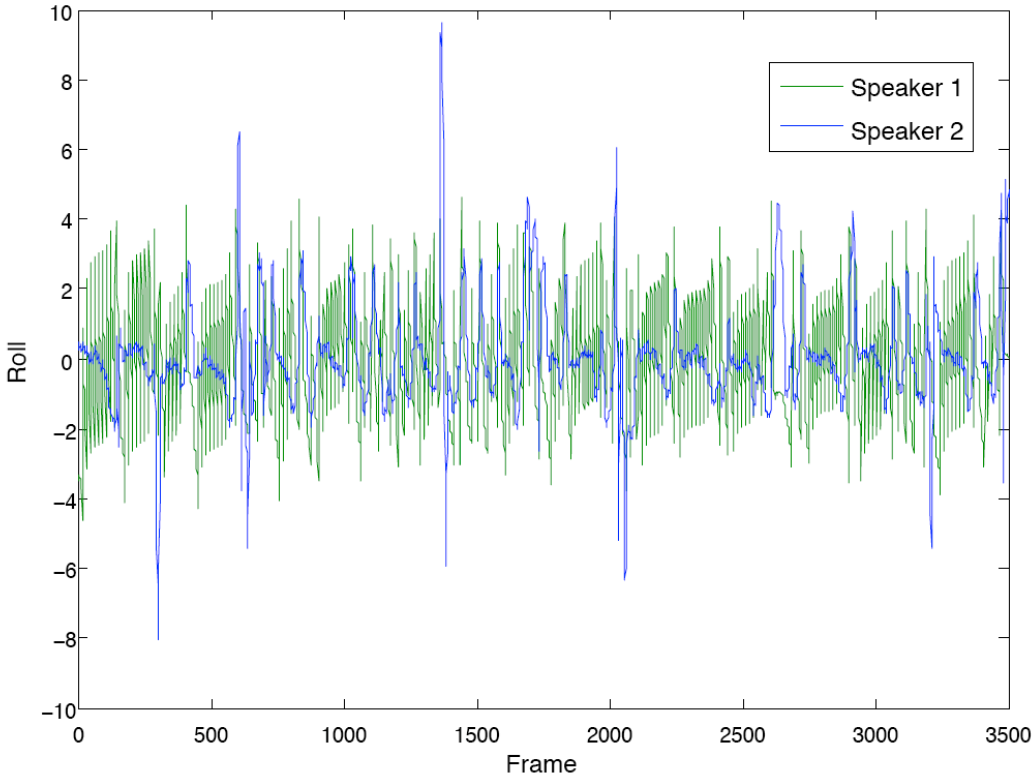

Figure 6.27: Head motion generated from the same label sequence for two different model sets. Each model set was trained on a different speaker

Features	Speech	Speech + lip
Accuracy	68.31	71.2

Table 6.10: Prediction accuracy for models using speech features and combined lip and speech features.

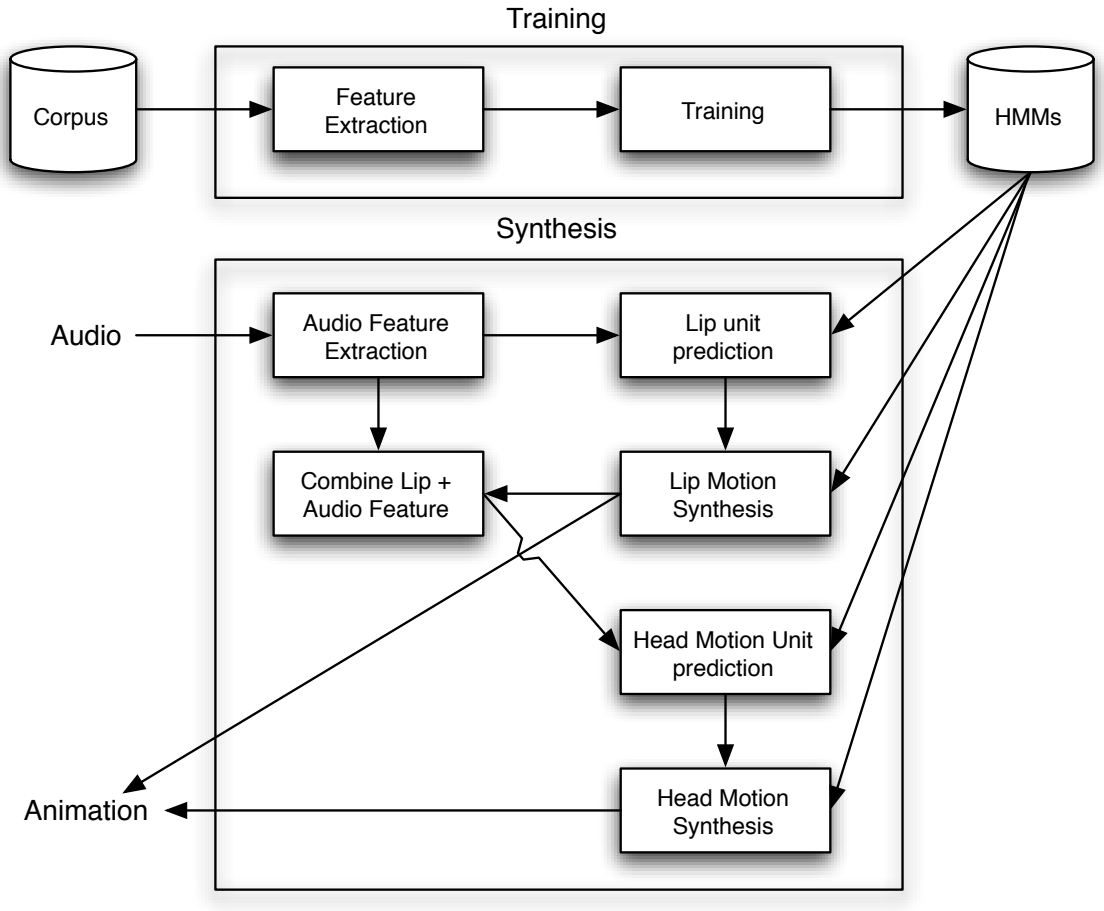

Figure 6.28: System diagram for an extension of the mapping method that takes lip motion into account. The HMMs are trained on speech, lip, and head motion features. During Synthesis, lip motion is predicted from the speech and synthesised. The generated lip trajectories are then combined with the speech motion features and used to predict head motion model. This

animation with each stage performing two steps; a recognition step and a synthesis step where the same kind of model is used for both steps. Each type of data (Lip,Head,...) is synthesised by a separate stage, and the output of that stage is fed into the next one, and so forth. In the recognition step we choose the most likely unit sequence given some speech. During synthesis the unit sequence is translated into motion trajectories. These two steps are performed for each type of motion data. In this case, lip motion and head motion are synthesised from speech data. Figure 6.28 gives a graphical overview of the process.

The models are trained on speech-derived features and motion capture data simulta-

neously using the maximum-likelihood criterion. Each type of data is modelled in a separate stream where only the transition probabilities between states are shared with the other streams. In our particular system, lip motion and eyebrow motion are treated in the same data stream and head motion in a different data stream. These streams are turned on and off depending on synthesis or recognition. For example, when predicting lip units the stream that models the lip motion trajectories is turned off and only the speech stream is turned on. Whereas during synthesis the speech stream is turned off and the motion stream is turned on. The parameter generation algorithm uses the predicted units, meaning that each unit corresponds to a model, to synthesise a smooth trajectory.

During synthesis time, speech data is recognised by the model and lip motion units are determined using the trained HMM. Trajectories are generated from the HMMs using the units. The lip motion trajectories are combined with the input speech and used to determine the head motion units. As previously confirmed this feature combination is beneficial. Then the head motion trajectories are generated from these units. Finally, both the head motion and lip motion trajectories are used to drive control points on our facial model.

6.9 Summary & Conclusion

This chapter described a method for generating speech-synchronised animation parameter sequences. The method is highly dependent on the employed modelling unit. Both a speech-based unit, namely phrases, and a motion-based unit were examined. Recognition experiments were carried out to find an optimal unit. The results of the perceptual evaluation of the different models trained on speech based and motion based units will be presented in the next chapter.

It is interesting to note that head motion types can be successfully predicted from speech features. Although the models performed the best when distinguishing between regions of high and low activity, they are still capable of distinguishing between types of motion. This lends support to the theory that head motion and speech are closely related and maybe part of the same physical process. Adding lip features improves the recognition accuracy, which makes sense as the lip motion and head motion are probably controlled by similar systems and more correlated than speech features and

head motion.

For the synthesis, the effect of synchrony seems to be better when the prediction was good. It is difficult to measure the degree synchrony between speech and motion specifically but preliminary perceptual evaluation suggests that the movement is synchronised with the rhythm of the speech. Both the dynamic range control and the non-deterministic synthesis make it possible to generate more interesting movements. As one of the problems with using a small number of units is that there is very little variation. However, in animation it is very important to be able to control the dynamic range. Having control over the dynamic range makes the system more useful to animators and gives it more credibility as something that could be used outside an academic setting.

Chapter 7

Perceptual Evaluation

7.1 Introduction

This chapter describes the perceptual evaluation carried out to validate the methods described in the previous chapter. For synthesis, it is important to have human judgements, as they are the ultimate consumers of the developed systems. Before the evaluation is described, the influence of the appearance of the characters is considered. Then the perceptual evaluation of the developed synthesis system is described. Finally, the results are presented.

7.2 Degrees of human-likeness

Humans have developed specialised processing for face recognition. Furthermore, people are able to judge attractiveness of others at a glance and are able to identify potential mates using markers for reproductive fitness. Overall, humans seem to hard-wired to deal with facial stimuli on various levels. When carrying out an evaluation of talking heads one has to be aware of the sensitivity of subjects to facial expressions.

One popular theory that can be applied to our inherent predisposition to judge faces has been described as the "Uncanny Valley of Eeriness" (Mori 1970). In 1970, Dr. Masahiro Mori described the following thought experiment: As robots become more and more similar to human form, would our familiarity rise with the human likeness or

would there be a more complex relationship between the two quantities. He hypothesises that familiarity increases with human-likeness up to a point and then drops, rising again when human-likeness comes closer to a healthy person, thus the graph describes a valley. Figure 7.1 shows the original graph. In addition Mori thinks that movement accelerates the effect.

The uncanny valley can be explained in part by biological and cognitive theories of aesthetics. The former are based on the biological basis of the perception of attractiveness. Members of different cultures show agreement on attractiveness ratings, and even babies show a preference towards attractive people. The judgement of reproductive fitness, using indicators of fertility, with attractive people displaying higher ratings on markers like skin quality, symmetry, and facial proportions, is almost instantaneous. Therefore perceiving someone as unattractive and lacking reproductive fitness, could be one explanation of the feelings of aversion associated with the uncanny valley (MacDorman et al. 2009, MacDorman & Ishiguro 2006, MacDorman 2005, Pollick to appear, Seyama & Nagayama 2007).

Although the uncanny valley is often presented as a fact-based theory, there is little evidence to support the right part of the graph, which shows familiarity rising again. Still, the uncanny valley plays a useful role when talking about the perception of synthetic characters as it points to the problem that small errors in the presentation of characters can have powerful effects on the perceiver. Similarly, the perception of moving characters needs to be considered in a similar fashion, where the relationship between the realism of the presentation and the realism of the movement might interact, creating a mismatch that degrades the overall impression of the character.

More research-based work argues that the effectiveness of information delivery by talking heads is highly correlated with the presentation of the character (Haddad & Klobas 2003). There is disagreement as to wether more cartoon-like characters are more successful or if photo-realism delivers information more effectively (Prendinger et al. 2005). One argument is that photo-realism creates higher expectations of the behaviour of the character and therefore might disappoint the user (Ellis & Bryson 2005). Animators outside the virtual agent community have shied away from photo-realism because of these problems.

Therefore this evaluation is aiming at alleviating the effect appearance has on the perception of the talking head. In addition to different movement synthesis methods,

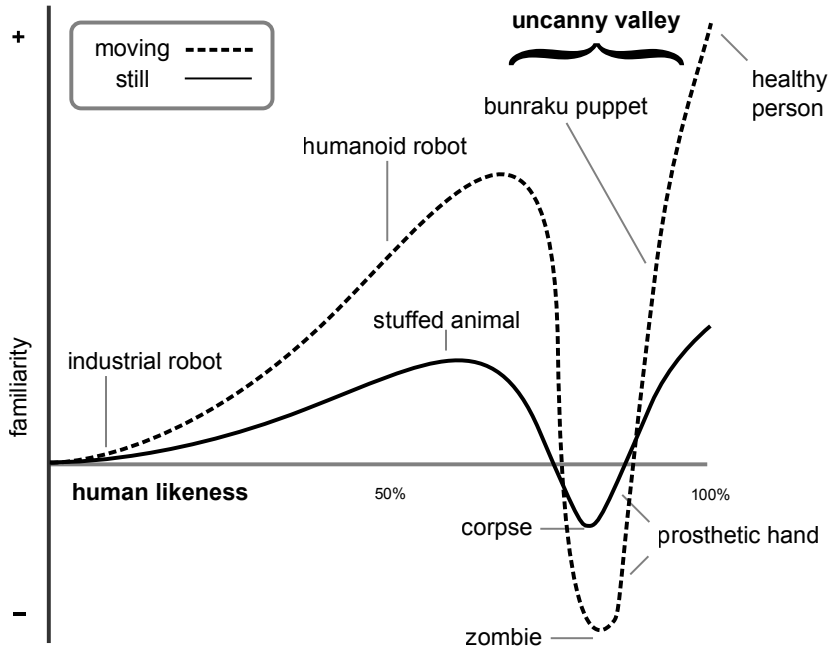

Figure 7.1: In 1970, the roboticist Masahiro Mori (Mori 1970) graphed a relationship between human-likeness and perceived familiarity. His hypothesis is that familiarity increases with human-likeness up to a point when subtle differences in appearance create a negative effect called the uncanny valley. Graph from MacDorman et al. (2009)

several different rendering qualities are investigated as well.

7.3 Perceptual Evaluation

7.3.1 Hypotheses

The first aim of the of the perceptual evaluation was to determine how the developed method for synthesising head motion performs in comparison to original motion capture playback and to a simple baseline.

H1A. The developed synthesis method performs better than the baseline.
H1B. The synthesised movement is rated as favourably as the original motion capture playback.

Another aim is to compare the standard synthesis method with the non-deterministic method that uses random mixture selection.

H2. The non-deterministic method is rated higher than the deterministic method.

Given the above hypotheses, a perceptual evaluation was carried out to confirm them. One very important aspect in the perception of virtual characters is the interaction between movement and appearance. Using the uncanny valley theory as a guide, this thesis is aiming to confirm this interaction. Therefore a final hypothesis is formulated.

H3 There is an interaction between rendering quality and movement quality.

7.3.2 Method

7.3.2.1 Subjects

The experiment was advertised on student mailing lists and 52 subjects took the test. Only 13 female subjects were recruited, with the rest being male. Only one subject indicated that he was a facial animation expert.

7.3.2.2 Condition

The evaluation tested the two independent variables, rendering quality and movement quality. Rendering Quality was divided into four categories using the FireFly (Smith-Micro 2008) rendering engine implemented in the Poser animation environment. Table 7.1 gives an overview of the rendering conditions.

Movement quality was alternated between four conditions. Table 7.2 gives an overview of the movement types.

Name	**Description**	**Example**
Plain	Single colour (beige) ray tracing render	
Textured	Textured ray tracing render	
Cartoon Color	Color cartoon shader render	
Cartoon BW	Black and white cartoon shader render	

Table 7.1: Four different rendering conditions used in the evaluation.

Name	**Description**
Baseline	Random head nods, with no movement during pauses in the speech
Motion Capture	Original motion capture played back
Deterministic	Deterministic Synthesis method using trajectory HMMs
Stochastic	Stochastic Synthesis method using trajectory HMMs, using random mixture selection

Table 7.2: Four different animation synthesis conditions used in the evaluation.

7.3.2.3 Procedures

A 3D model was developed for the evaluation. The mesh is a full body model of a male human, with an underlying bone system. In addition to the bones it is possible to control the mouth using a set of morph targets. The control of the animation is done using Poser (SmithMicro 2008), a commercial 3D animation package. Although Poser is not capable of producing real time animation, it has powerful scripting capabilities. Distributed rendering is possible by rendering each frame of an animation as a separate process.

A Flash-based webpage was created in order to carry out the evaluation. Participants were asked to provide their first name, gender, and their familiarity with facial animation. Once they clicked start, the experiment began. An example of the stimulus presentation page is given in Figure 7.2. Videos could be played by clicking on the thumbnails at the bottom of the page. The video that is currently playing is indicated by a blue border around the thumbnail. Each video could be watched as many times as they wanted, but each one had to be played at least once in order to move to the next page. The average length for each video was about 8 seconds. All the animations came form models trained on speaker 1 and consisted of sentences that were not seen in the training data. The videos were rated by indicating the best and the worst fit between movement quality and image quality. A description on how to play the videos and how to rate them was given at the right side of the page. The evaluation specifically asked for the fit between image quality and movement quality and did not define naturalness as subjects were supposed to focus on the fit.

The presentation of stimuli was randomised over both conditions, with all subjects seeing all possible combinations. Each combination was presented five times. In total there were 20 pages with 4 videos each, with each subject rating 80 videos.

7.3.2.4 Results

The results were evaluated by attributing a score of 1 if a subject selected a best video, score of -1 for worst video, and 0 if the video was not selected at all. A pie chart for the percentage of scores in each rendering condition is presented in Figure 7.3. From that it can be seen that subjects seem to prefer the deterministic synthesis over the other synthesis methods. Although the stochastic synthesis is preferred in the ‘Textured’

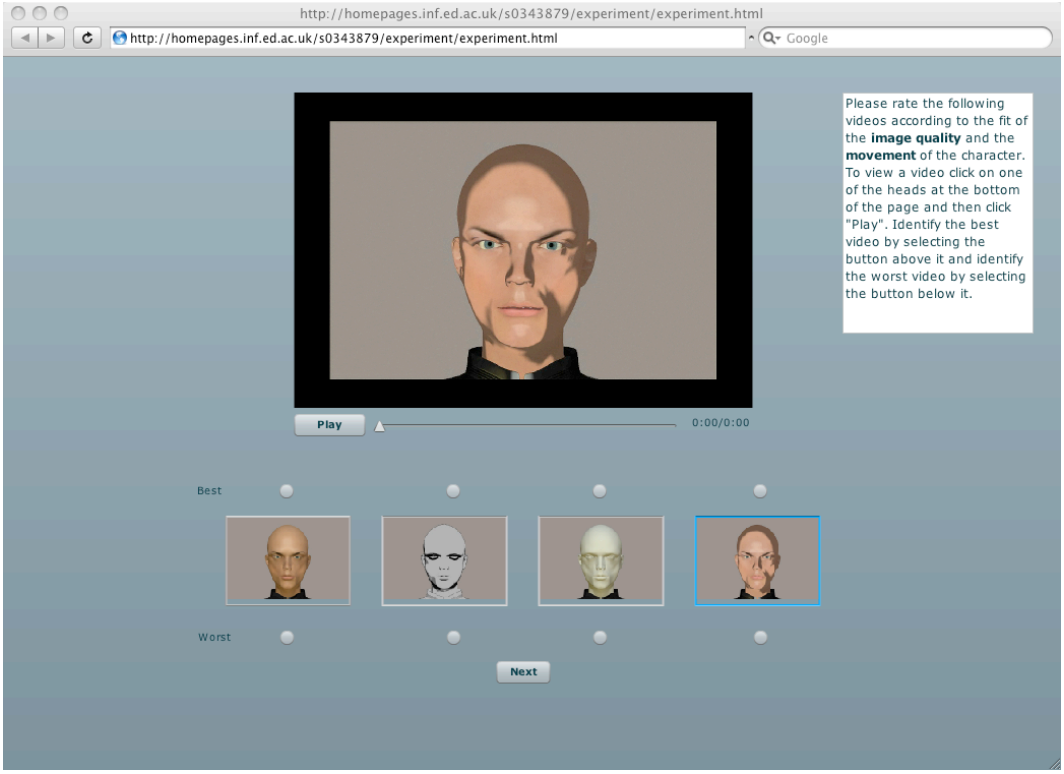

Figure 7.2: Each page of the experiment featured four videos that could be played by clicking the thumbnail at the bottom of the page. Subjects could rate them by selecting on of the radio buttons on top of the thumbnails, indicating the best video, and likewise selecting one of the radio buttons on the bottom, indicating the worst video.

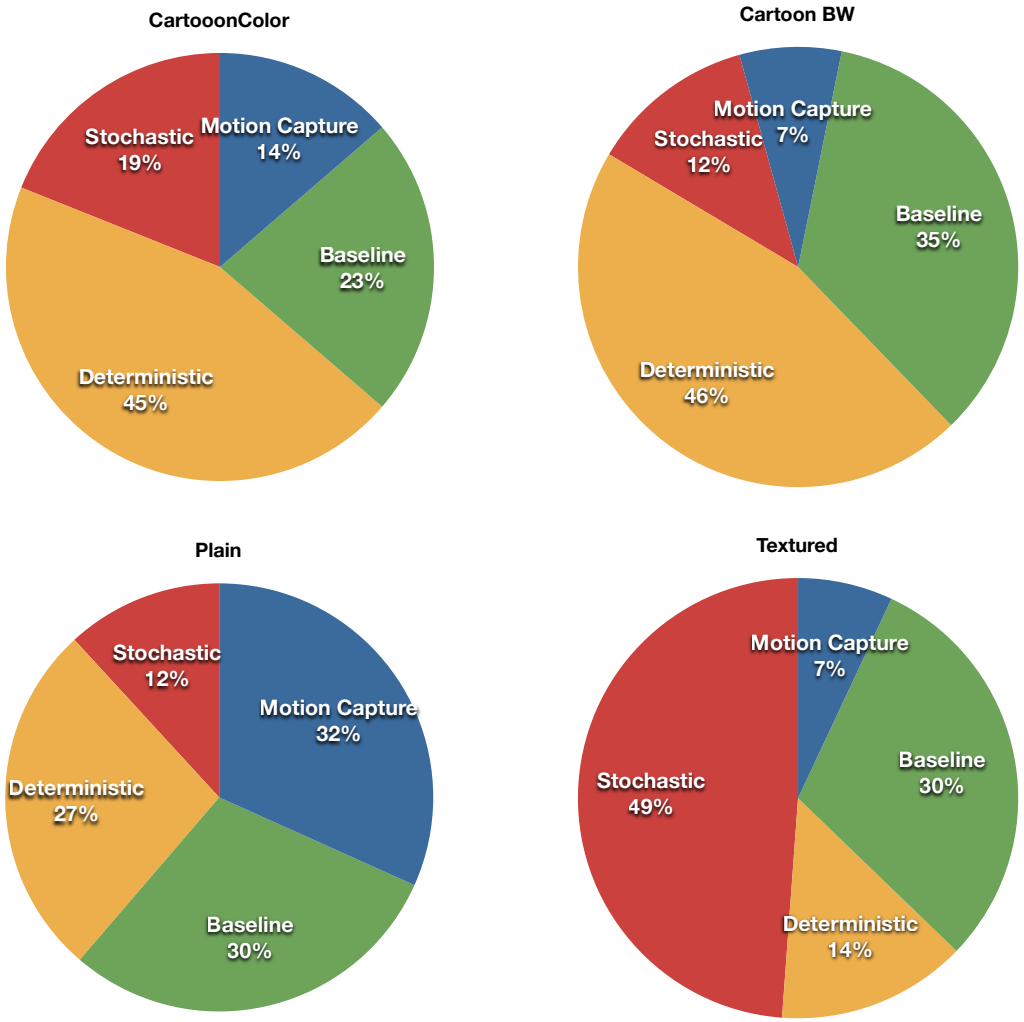

Figure 7.3: Pie chart showing the percentage of subjects that preferred each animation synthesis condition in the different rendering conditions. Deterministic synthesis seems to be the most preferred one, when people liked the rendering. For the conditions where people did not like the rendering the results are not as clear cut.

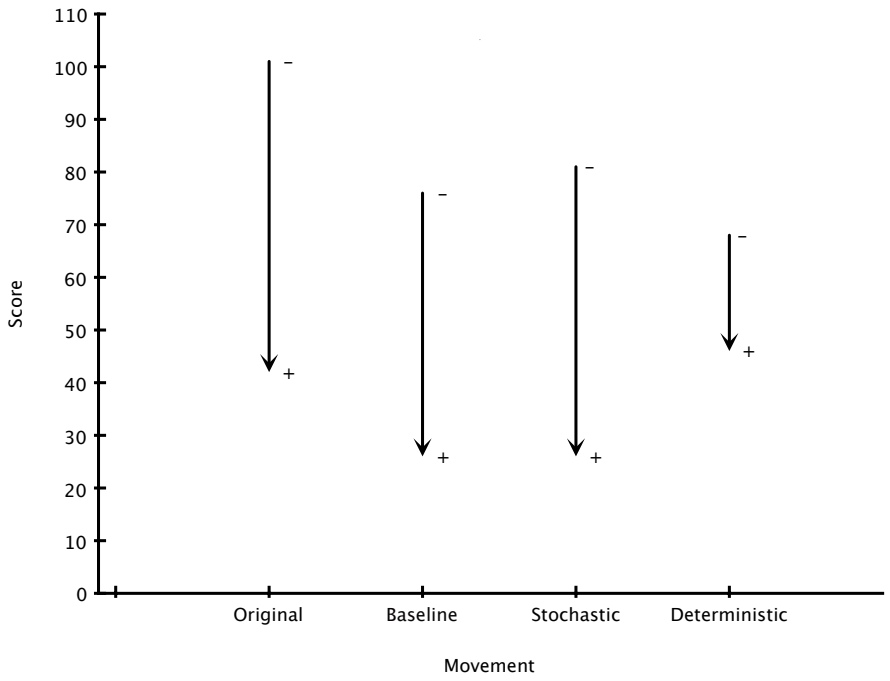

Figure 7.4: PLain results: The number of 'best' responses is marked with + and the number of 'worst' responses is marked with -. The length of the arrow indicates if the confidence of subjects in the decision. A longer arrow means higher confidence.

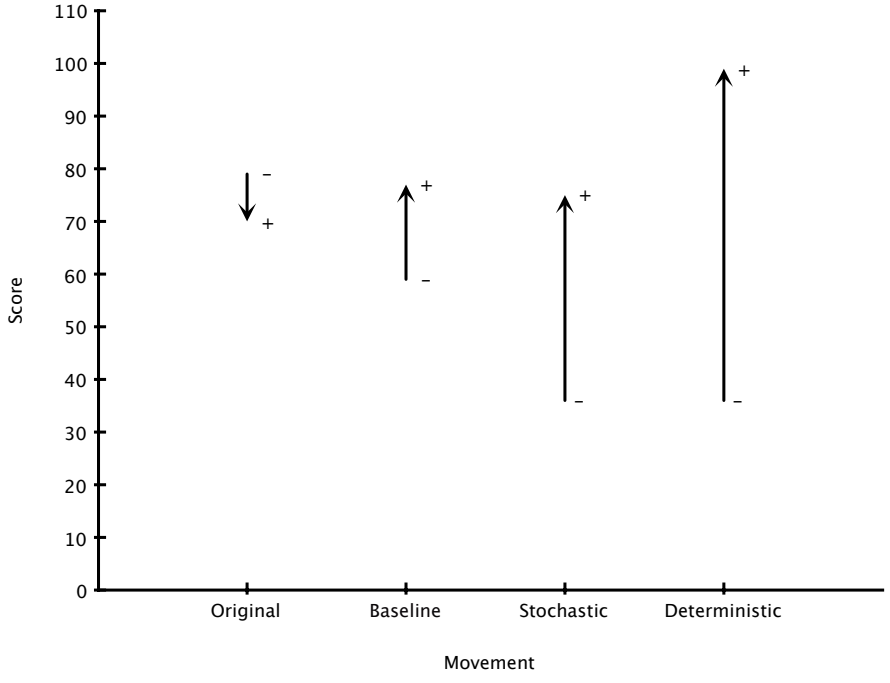

Figure 7.5: Textured results: The number of 'best' responses is marked with + and the number of 'worst' responses is marked with -. The length of the arrow indicates if the confidence of subjects in the decision. A longer arrow means higher confidence.

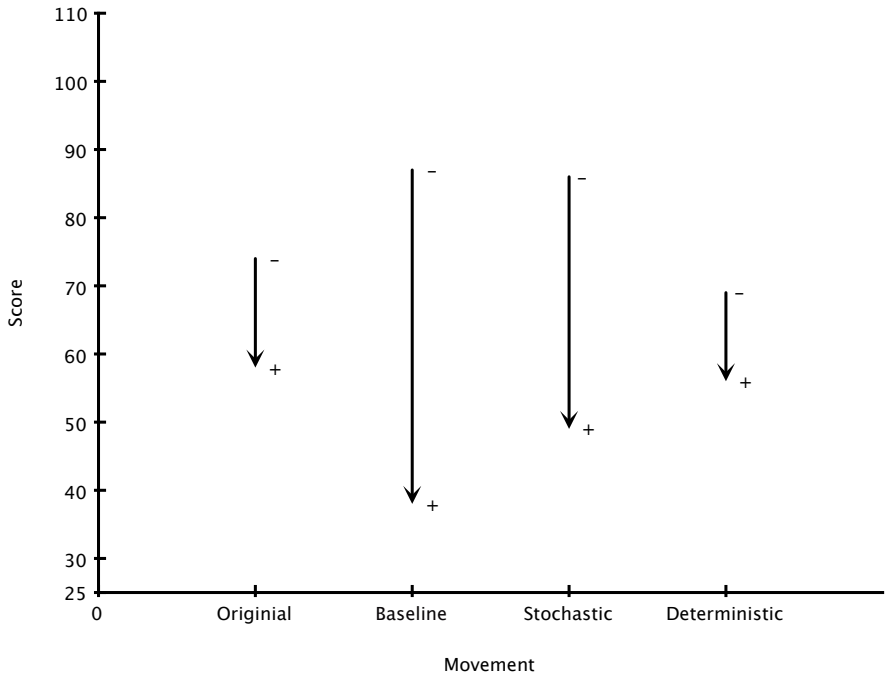

Figure 7.6: CartoonBW results: The number of 'best' responses is marked with + and the number of 'worst' responses is marked with -. The length of the arrow indicates if the confidence of subjects in the decision. A longer arrow means higher confidence.

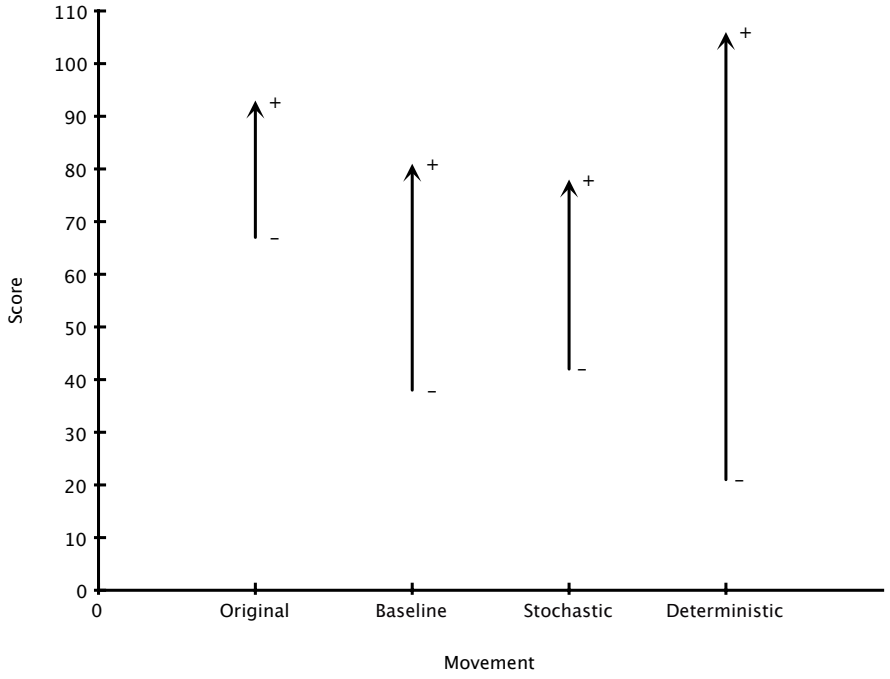

Figure 7.7: CartoonColor results: The number of 'best' responses is marked with + and the number of 'worst' responses is marked with -. The length of the arrow indicates if the confidence of subjects in the decision. A longer arrow means higher confidence.

	Motion capture	Stochastic	Baseline	Deterministic
Motion capture		$p = 0.2069$	$p = 0.5395$	$\mathbf{p < 0.00001}$
Stochastic			$p = 0.4967$	$\mathbf{p < 0.00001}$
Baseline				$\mathbf{p < 0.00001}$
Deterministic				

Table 7.3: P-values from the t-tests comparing the means for each synthesis condition, significant differences $(p < 0.01)$ are printed in bold. The deterministic synthesis seems to outperform the other methods.

	cartoonBW	cartoonColor	textured	plain
cartoonBW		$\mathbf{p < 0.00001}$	$\mathbf{p < 0.00001}$	$p = 0.0204$
cartoonColor			$p = 0.01303$	$\mathbf{p < 0.00001}$
textured				$\mathbf{p < 0.00001}$
plain				

Table 7.4: P-values from the t-tests comparing the means for each rendering condition, significant differences $(p < 0.01)$ are printed in bold.

rendering condition. Another way of looking at the data where the ratio of best and worst responses are plotted for each rendering condition shown in Figs. 7.4, 7.5, 7.6, and 7.7 respectively, confirmed that subjects preferred the deterministic synthesis. A t-test was performed comparing the movement synthesis conditions with each other. The results can be seen in Table 7.3. The Deterministic synthesised movement was preferred significantly over the other synthesised variants.

A t-test was performed comparing the rendering conditions with each other. The results can be seen in Table 7.4. The coloured cartoon rendering method was preferred overall when compared directly with the other rendering methods. Subjects disliked the plain rendering condition most often.

A Factorial Analysis of Variance (ANOVA) was performed to see if there are significant interactions between rendering quality and movement. An ANOVA is a statistical test of whether the means of several groups are equal. In can be generalised to more than one independent variable, where each has two or more distinct values. The independent variables in this case are movement quality (motion capture, stochas-

Source	df	Mean Sq.	F value	Significance
render	3	31.45	67.52	***
movement	3	5.84	12.43	***
render:movement	9	0.91	1.96	*

Table 7.5: Anova summary, significance is indicated with * ($p < 0.05$) and *** ($p < 0.01$)

tic generation, baseline, deterministic generation) and rendering quality (cartoon BW, cartoon color, textured, plain). The variables can have different values at different combinations, so "plain:motion capture" has a different score from "plain:baseline". The ANOVA tests for significant interaction of the combination of these two variables. The results of the two way ANOVA are shown in Table 7.5.

7.3.2.5 Summary

The hypothesis that the developed system performed better than the baseline was confirmed (H1A) in Table 7.3. Additionally the synthesised movement was rated higher than the original motion capture playback (H1B) also shown in Table 7.3. Unfortunately the stochastic method performed significantly worse than the deterministic method (H2) as confirmed in in Table 7.3. There were no significant differences found between the stochastic method and the baseline or the original motion capture. Finally, concerning the rendering quality, the results shown in Table 7.5 confirm a significant interaction between rendering quality and movement quality (H3).

7.4 Discussion

Overall, the deterministic synthesis was preferred over the stochastic one. In our system the stochastic synthesis was less smooth than the deterministic one, because the mixture component selection was done frame wise, where state wise selection on the other hand might have lead to smoother results. A new mixture component was selected every frame, but if one mixture component would have been selected for each state, the output might have been smoother. Although, overall the deterministic syn-

thesis was preferred, in the 'Textured' condition the stochastic synthesis was preferred. It is not clear why this is so but it might have to do with how the smoothness of the movement was perceived under different rendering conditions. The stochastic motion synthesis may have been less smooth than the deterministic overall, but the 'Textured' face might somehow hide the lack of smoothness in the stochastic movement condition. Still, it is possible that with longer utterances, the stochastic movement might have been preferred overall. The utterances in the evaluation were quite short, and therefore there was little repetition of movements. Longer utterances, on the other hand, would produce similar motion over and over again. In this case the stochastic generation could fare better as it can generate more interesting movements.

In the experiment the synthesised version of the head motion was rated higher than the original playback of the head motion, which seems counter-intuitive. There are several reasons why this could be. It is not straightforward to replay motion capture data, as it was recorded from a real human, and playing it back requires retargeting the trajectories to the model that plays it back, be it a ball or a photorealistic head. Therefore motion capture data does not necessarily look "natural" when played back on a 3D mesh. Furthermore, the dynamic range of the original movement was quite large, as the speaker was moving quite erratically during recording. The dynamic range of the other movements was smaller due to statistical modelling, which might be more appropriate for an artificial character. Finally, people might just like the computer generated movement better than the original because it was displayed on a 3D mesh.

In terms of the rendering quality conditions the outcome of the experiment was clear. The 'cartoonColor' rendering condition was preferred overall, followed by the 'Textured' version. The lower quality rendering conditions, 'plain' and 'cartoonBW' were not preferred by many subjects. In the future it is clear that the quality of the appearance of the character needs to varied in a more principled way. For this study, four representative types were picked by the experimenter but the dimensions of appearance need to be more carefully defined to be able to determine the type of interaction between appearance and movement.

Finally, the hypothesis was confirmed that there is an interaction between movement and rendering quality. The direction of the interaction between these two factors is not entirely clear but it seems that the more realistic a face appears the higher expectations the user has about the behaviour of the face. There was support for this hypothesis

because the gap between the baseline and the synthesis methods was narrower for the lower quality faces. However, the opposite hypothesis that lower quality movement is preferred with lower quality faces was not explicitly stated, subjects do rate the motion captured movement lower as the synthesised ones, hinting that there is an 'uncanny' effect in the opposite direction. Overall, the results lend more evidence to the theory that using more photorealistic faces does not necessarily mean that users perceive them as more positive. The relationship of the movement quality and the rendering quality has to be described in more detail before real predictions can be made, but it is clear now that when designing a system, one has to consider the realism of the face and realism of the movement together.

Chapter 8

Discussion & Conclusion

The main goal of this thesis was to demonstrate that it is feasible to generate head motion from speech. The specific contributions made were:

1. It was shown that it is feasible to generate head motion from speech without linguistic analysis.
2. A human-readable unit of head motion was developed and investigated.
3. A stream mapping method that is capable of mapping between speech and head motion was developed.
4. The application of trajectory HMM-based synthesis to motion synthesis was performed.
5. The MLE parameter generation algorithm was extended to allow for non-deterministic synthesis.

The reason that head motion can be generated from the speech signal can be found where speech synchronised motion originates. Head motion in itself is hypothesised to consist of two different types of movement: motoric and linguistic. The distinction can be made on the axis of awareness, where we are almost unaware of the motoric movement but quite aware of the linguistic movements. Humans are aware of functional movements because they are usually performed to express some meaning, like pointing at an object or nodding in agreement. Motoric movements on the other hand are part of the articulation, which do not have any semantic relationship to what is being said but just support the movement of the articulators. We are mostly unaware

of these kinds of movements. The head motion generation done in this thesis focused on this type of motion as it can be predicted from the speech signal. It was shown that types of motion can be distinguished by analysis the speech signal. It is therefore possible to predict head motion types from speech features.

Furthermore, another element of this thesis was the development of a stream mapping method that is able to synthesise a motion stream given a speech stream. Various mapping possibilities were proposed such as graphical models and a unit based mapping technique, and investigated as part of this thesis. Much time was spent finding the optimal unit as the mapping method depended on it. As an optimality criterion, the accuracy of the prediction was used, which was a straightforward objective measure. One interesting finding was that the models performed better with more states than the standard speech recognition HMMs. Additionally it was found that models that combined speech and head motion features outperformed HMMs that were trained on only one stream.

The trajectory HMM was employed for the trajectory synthesis part. In its original form it is not possible to synthesise stochastic trajectories since the MLE parameter generation algorithm is deterministic. Although this might be sufficient for speech synthesis, in motion synthesis there is more than one way of achieving movement that looks natural, especially in speech-related movement, which is synchronised with what is being said. The rhythm of the speech can be expressed in many different ways. Therefore the MLE parameter generation algorithm was extended to allow for non-deterministic synthesis. In the final perceptual evaluation this extension did not fare as well as the standard algorithm. There are two reasons for this. First, there are still some technical issues with the stochastic parameter generation, where some generated trajectories have large frame-wise changes, which makes the motion look "jerky". The mixture component selection works frame-wise, meaning that each frame a new mixture component is chosen. This could be changed to state-wise mixture component selection, meaning that once a mixture component is chosen randomly for a given state, that mixture component is used for as long as parameters are generated from that state. Second, the utterances in the evaluation were short (< 10 seconds) and we believe that if longer utterances or full paragraphs were judged, the non-deterministic method would fare better, because there would be less repetitive behaviour.

Another possible applications of the developed non-deterministic trajectory generation algorithm is prosody generation for speech synthesis. One of the drawbacks of modern speech synthesis techniques is the predictability of the prosody, employing a non-deterministic synthesis method might make listening to synthesised speech more pleasant.

During evaluation, the problem of the "Uncanny Valley", which was described in detail in Chapter 7, became more apparent. When judging motion of a character, the look of the character has a large influence on our perception and judgement. We are perceiving the whole situation, not just the movement of the character and therefore the appearance and movement have to fit. The perceptual evaluation carried out on the developed system, attempted to negate the effect of appearance by having people judge the same movement on characters that were rendered in different qualities. The suspicion was confirmed that there is an interaction between quality of movement and quality of rendering. What was interesting is that people preferred cartoon-like looking characters over more realistic rendered characters. This has been known and exploited in the film and game industry, which has shied away from using photorealistic characters, because one can get an uncanny effect from them. It is easier to achieve satisfactory results with cartoon characters. This was confirmed also in the evaluation carried out during this thesis. The hypothesis was that rendering quality and movement quality should be on the same level. That means that highly realistic rendering requires the movement to be realistic but also vice versa that less realistic rendering requires less realistic movement. It was confirmed that realistic rendering requires the animation to be more realistic but the opposite, that less realistic rendering requires less realistic animation could not be confirmed. Realistic movement is still preferred over less realistic movement even if the character is rendered in lower quality. One possible reason for this could be that the underlying mesh was the same for all characters and maybe the realism of the character as a whole needs to be more cartoon like. However, more research into the exact interaction between movement quality and appearance is needed.

To conclude, the proposed mapping method is general and could be employed for other speech-related motion, like hand gestures or eyebrow movements. The method has already been shown to work in an integrated way for lip motion and head motion. Although there is an advantage to using hand labelled units, since they are hu-

man readable, better methods for automatically determining units should be developed. Nevertheless the use of clustering methods in determining units was investigated. Sequence clustering remains a difficult problem and more sophisticated methods based on probabilistic modelling could be employed in the future.

Appendix A

Phoneme 2 Viseme Map

The following table gives the mapping from the phone set to the different viseme sets that were employed in the lip synchronisation.

CMU Phone	Preston Blair	eVis	2Vis	CMU Phone	Preston Blair	eVis	2Vis
@	k	c	o	n!	k	c	o
a	a	a	o	ng	k	l	o
aa	a	a	o	o	o	o	o
ai	a	ai	o	oi	o	oi	o
b	m	m	c	oo	o	oo	o
ch	k	c	o	ou	o	oo	o
d	k	c	o	ow	o	ow	o
dh	k	c	o	p	m	m	c
e	e	e	o	r	k	c	o
ei	e	ei	o	@@r	k	c	o
eir	e	a	o	s	k	k	o
f	f	f	o	sh	k	c	o
g	k	c	o	t	k	th	o
h	k	c	o	th	k	th	o
i	a	e	o	u	u	u	o
i@	k	c	o	uh	u	u	o
ii	e	e	o	ur	u	u	o
iy	e	e	o	uu	u	u	o
jh	k	c	o	uw	u	uw	o
k	k	c	o	v	f	f	o
l	l	l	o	w	q	q	o
l!	l	l	o	y	k	c	o
lw	l	l	o	z	k	c	o
m	m	m	c	zh	k	c	o
n	k	c	o				

Table A.1: The mapping from the CMU phone set to the Preston Blair, eVis, and 2Vis viseme sets.

Appendix B

3D Character Animation

Historically, animation is usually divided into 2D and 3D animation. Both fields have distinct algorithms and software associated with each other. Although, the methods developed in this thesis are applicable to either 2D or 3D animation, it focuses only on 3D animation, because 3D motion capture data was used. Although this thesis focuses mainly on the modelling and generation of animation parameters, a brief description of the developed animation system is given.

B.1 Blend Shapes

Figure B.1: The sum of two blend shapes produces a mesh.

A mesh can be animated using blend shapes by combining or morphing meshes with the same topology. Joshi et al, Joshi et al. (2006) define a blend shape model as a convex linear combination of n basis vectors, each vector is a blend shape.

Formally a vertex V in the blend shape model is defined as

$$V = \sum_{i=1}^{N} \alpha_i v_i \tag{B.1}$$

where the scalars α_i, are the blending weights, v_i is the location of vertex in blend shape i, and N is the number of blend shapes. Also $alpha$ must satisfy the constraints:

$$\alpha_i \leq 0 \tag{B.2}$$

$$\sum_{i=1}^{N} \alpha_i = 1 \tag{B.3}$$

B.2 Skin and Bones

Skeletal animation is based on the idea that the mesh is transformed via an underlying transformation matrix set. Where each vertex in the mesh is transformed via weighted transform for each bone. This is called mesh palette skinning, where skinning is the weighting of the transforms for each vertex.

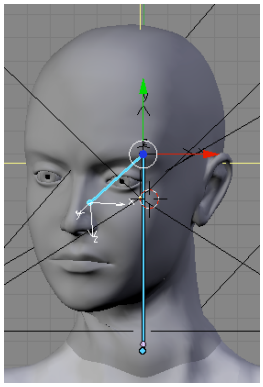

Figure B.2: A talking head with the underlying neck bone structure.

The transform has the following components:

- K . . . number of matrices
- v . . . vertex position
- β_i . . . weight
- M_i . . . transformation matrix

Each vertex v is transformed by a weighted sum of transforms, e.g.

$$v' = \sum_{i=1}^{K} \beta_i M_i v \tag{B.4}$$

where (B.5)

$$\sum_{i=1}^{K} \beta_i = 1 \tag{B.6}$$

Bibliography

AB, Q. (2006), *Qualisys Track Manager: User Manual*.

Aleksic, P. S. & Katsaggelos, A. K. (2004), 'Speech-to-video synthesis using MPEG-4 compliant visual Features', *IEEE Transactions on Circuits and Systems for Video Technology* **14**(5), 682–692.

Bailly, G., Bérar, M., Elisei, F. & Odisio, M. (2003), 'Audiovisual speech synthesis', *International Journal of Speech Technology* **6**, 331–346.

Bengio, Y. & Frasconi, P. (1995), 'Input/output HMMs for sequence processing', *IEEE Transactions on Neural Networks* **7**, 1231–1249.

Birdwhistell, R. (1970), *Kinesics in Context*, University of Pennsylvania Press.

Borga, M. (2001), 'Canonical correlation: a tutorial', *Online tutorial http://people.imt.liu.se/magnus/cca* .

Brand, M. (1999), Voice puppetry, *in* 'SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques', ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 21–28.

Brand, M. & Hertzmann, A. (2000), Style machines, *in* 'SIGGRAPH '00: Proceedings of the 27th annual conference on Computer graphics and interactive techniques', ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, pp. 183–192.

Brand, M. & Shan, K. (1998), 'Voice-driven animation', *Proc. Workshop on Perceptual User Interfaces* .

Busso, C., Deng, Z., Grimm, M., Neumann, U. & Narayanan, S. (2006), 'Rigid head motion in expressive facial animation: Analysis and synthesis', *IEEE Transaction on Audio, Speech and Language Processing* **15**(3), 1075–1086.

Cao, Y., Tien, W. C., Faloutsos, P. & Pighin, F. (2005), 'Expressive speech-driven facial animation', *ACM Transactions on Graphics* **24**(4), 1283–1302.

Cassell, J., Pelachaud, C., Badler, N., Steedman, M., Achorn, B., Becket, T., Douville, B., Prevost, S. & Stone, M. (1994), Animated conversation: rule-based generation of facial expression, gesture & spoken intonation for multiple conversational agents, *in* 'SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques', ACM, New York, NY, USA, pp. 413–420.

Cassell, J., Vilhjálmsson, H. H. & Bickmore, T. (2001), Beat: the behavior expression animation toolkit, *in* 'SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques', ACM, New York, NY, USA, pp. 477–486.

Chang, Y.-J. & Ezzat, T. (2005), Transferable videorealistic speech animation, *in* 'SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation', ACM, New York, NY, USA, pp. 143–151.

Chen, T. (2001), 'Audiovisual speech processing', *Signal Processing Magazine, IEEE* **18**(1), 9–21.

Choi, K., Luo, Y. & Hwang, J.-N. (2001), 'Hidden markov model inversion for audio-to-visual conversion in an mpeg-4 facial animation system', *Journal of VLSI Signal Processing Systems* **29**(1/2), 51–61.

Cohen, M. M., Massaro, D. W. & Clark, R. (2002), Training a talking head, *in* 'Proceedings of the 4th IEEE International Conference on Multimodal Interfaces', IEEE Computer Society, Washington, DC, USA, p. 499.

Cohen, M. & Massaro, D. (1990), 'Synthesis of visible speech', *Behavioral Research Methods, Instrumentation and Computers,* **22**, 260–263.

Cohen, M. & Massaro, D. (1993), 'Modeling coarticulation in synthetic visual speech', *Models and Techniques in Computer Animation* .

DeCarlo, D., Revilla, C., Stone, M. & Venditti, J. (2002), Making discourse visible: Coding and animating conversational facial displays, *in* 'Proceedings of Computer Animation 2002', pp. 11–16.

Dittmann, A. T. & Llewellyn, L. G. (1969), 'Body movement ans speech rhythm in social conversation', *Journal of Personality and Social Psychology* **9**(2), 98–106.

Ellis, P. M. & Bryson, J. J. (2005), The significance of textures for affective interfaces, *in* 'Lecture Notes in Computer Science', Springer-Verlag, London, UK, pp. 394–404.

Ezzat, T., Geiger, G. & Poggio, T. (2002), 'Trainable videorealistic speech animation'.

Graf, H., Cosatto, E., Strom, V. & Huang, F. J. (2002), Visual prosody: facial movements accompanying speech, *in* 'Automatic Face and Gesture Recognition, 2002. Proceedings. Fifth IEEE International Conference on', pp. 396–401.

Hadar, U., Steiner, T., Grant, E. & Rose, F. (1983), 'Head movement correlates of juncture and stress at sentence level', *Language and Speech* **2**, 451–471.

Hadar, U., Steiner, T. & Rose, F. (1984), 'Involvement of head movement in speech production and its implications for language pathology', *Advances in Neurology* **42**, 247–261.

Hadar, U., Steiner, T. & Rose, F. C. (1985), 'Head movement during listening turns in conversation', *Journal of Nonverbal Behavior* **9**(4), 214–228.

Haddad, H. & Klobas, J. (2003), 'The relationship between visual abstraction and the effectiveness of a pedagogical character-agent', *Proceedings of AAMAS 2002 Workshop on Embodied Agents* .

Heylen, D. (2006), 'Head gestures, gaze and the principles of conversational structure', *International Journal of Humanoid Robotics* **3**(3), 241–267.

Hong, P., Wen, Z. & Huang, T. (2002), 'Real-time speech-driven face animation with expressions using neural networks', *IEEE Transactions on Neural Networks* **13**(4), 916–927.

Hsieh, C. & Chen, Y. (2006), 'Partial linear regression for speech-driven talking head application', *Signal Processing: Image Communication* **21**(1), 1–12.

Hsieh, C.-K. & Chen, Y.-C. (2005), Partial linear regression for audio-driven talking head application, *in* 'IEEE International Conference on Multimedia and Expo, 2005. ICME 2005.', pp. 281–284.

Huang, X., Acero, A. & Hon, H. (2001), *Spoken Language Processing*, Prentice Hall.

Joshi, P., Tien, W. C., Desbrun, M. & Pighin, F. (2006), Learning controls for blend shape based realistic facial animation, *in* 'SIGGRAPH '06: ACM SIGGRAPH 2006 Courses', ACM, New York, NY, USA, p. 17.

Kanwisher, N., McDermott, J. & Chun, M. (1997), 'The fusiform face area: a module in human extrastriate cortex specialized for face perception', *Journal of Neuroscience* **17**(1), 4302–4311.

Kendon, A. (2003), 'Some uses of the head shake', *Gesture* **2**(2), 147–182.

Kendon, A. (2004), *Gesture: Visible Action as Utterance*, Cambridge University Press.

Li, Y. & Shum, H.-Y. (2006), 'Learning dynamic audio-visual mapping with input-output hidden markov models', *IEEE Transactions on Multimedia* **8**(3), 542–549.

Ling, Z., Richmond, K., Yamagishi, J. & Wang, R. (2009), 'Integrating articulatory features into HMM-based parametric speech synthesis', *IEEE Transactions on Audio, Speech and Language Processing* **17**(6), 1171–1185.

MacDorman, K. (2005), Androids as an experimental apparatus: Why is there an uncanny valley and can we exploit it, *in* 'CogSci-2005 Workshop: Toward Social Mechanisms of Android Science', pp. 106–118.

MacDorman, K., Green, R., Ho, C. & Koch, C. (2009), 'Too real for comfort? uncanny responses to computer generated faces', *Computers in Human Behavior* **25**, 695–710.

MacDorman, K. & Ishiguro, H. (2006), 'Toward social mechanisms of android science', *Interaction Studies* **17**(3), 361–368.

Martin, G. (2006), 'Preston blair phoneme series', World wide web electronic publication.
URL: *http://www.canadianarachnology.org/data/spiders/30843*

McClave, E. (2000), 'Linguistic functions of head movements in the context of speech', *Journal of Pragmatics* **32**(7), 855–878.

McNeill, D. (2005), *Gesture and Thought*, The University of Chicago.

Mori, M. (1970), 'The uncanny valley', *Energy* **7**(4), 33–35.

Munhall, K., Jones, J., Callan, D. & Kuratate, T. (2004), 'Head movement improves auditory speech perception', *Psychological Science* **15**, 133–137.

Nakamura, S. (2002), 'Statistical multimodal integration for audio-visual speech processing', *Neural Networks, IEEE Transactions on* **13**(4), 854–866.

Pelachaud, C., Badler, N. & Steedman, M. (1996), 'Generating facial expressions for speech', *Cognitive Science: A Multidisciplinary Journal* **20**(1), 1–46.

Pollick, F. (to appear), Analog communication: Evolution, brain mechanisms, dynamics, simulation, *in* 'The Vienna Series in Theoretical Biology', MIT Press.

Prendinger, H., Mori, J. & Ishizuka, M. (2005), 'Recognizing, modeling, and responding to users' affective states', *Springer Lecture Notes in Computer Science* pp. 60–69.

Rabiner, L. (1989), A tutorial on hidden markov models and selected applications in speech recognition, *in* 'Proceedings of the IEEE', Vol. 77, pp. 257–286.

Rimé, B. & Schiaratura, L. (1991), Gesture and speech, *in* R. S. Feldman & B. Rimé, eds, 'Fundamentals of Nonverbal Behavior', chapter 7, pp. 239–281.

Sargin, M. E., Erzin, E., Yemez, Y., Tekalp, A. M., Erdem, A. T., Erdem, C. & Ozkan, M. (2007), Prosody-driven head-gesture animation, *in* 'Procedeedings of ICASSP 07', Vol. 2, pp. 677–680.

Seyama, J. & Nagayama, R. S. (2007), 'The uncanny valley: Effect of realism on the impression of artificial human faces', *Presence: Teleoperators and Virtual Environments* **16**(4), 337–351.

SmithMicro (2008), *Poser: Reference Manual*.

Tamura, M., Kondo, S., Masuko, T. & Kobayashi, T. (1998), Text-to-visual speech synthesis based on parameter generation from hmm, *in* 'Proceedings of ICASSP 98', pp. 3745–3748.

Toda, T. & Tokuda, K. (2007), 'A speech parameter generation algorithm considering global variance for HMM-based speech synthesis', *IEICE - Transactions on Information and Systems* **E90-D**(5), 816–824.

Tokuda, K., Yoshimura, T., Masuko, T., Kobayashi, T. & Kitamura, T. (2000), Speech

parameter generation algorigthms for HMM-based speech synthesis, *in* 'Proceedings of ICASSP 00', pp. 1315–1318.

Wen, Z., Hong, P. & Huang, T. S. (2001), 'Real time speech driven facial animation using formant analysis', *IEEE International Conference on Multimedia and Expo* **0**, 208.

Yehia, H., Kuratate, T. & Vatikiotis-Bateson, E. (2002), 'Linking facial animation, head motion and speech acoustics', *Journal of Phonetics* **30**(3), 569–590.

Young, S., Odell, J., Ollason, D. & Valtchev, V. (1995), 'The htk book', *Cambridge University* .

Zen, H., Tokuda, K. & Kitamura, T. (2007), 'Reformulating the hmm as a trajectory model by imposing explicit relationship between static and dynamic features', *Computer Speech* **21**(1), 153–173.

Zhang, S., Wu, Z., Meng, H. & Cai;, L. (2007), Head movement synthesis based on semantic and prosodic features for a chinese expressive avatar, *in* 'Proceedings of ICASSP 07', Vol. 4, pp. 837 – 840.